

## MPI Wrapper

Generated by Doxygen 1.9.1



<b>1 Namespace Index</b>	<b>1</b>
1.1 Namespace List	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 File Index</b>	<b>5</b>
3.1 File List	5
<b>4 Namespace Documentation</b>	<b>7</b>
4.1 MPIw Namespace Reference	7
4.1.1 Function Documentation	10
4.1.1.1 Allgather() [1/2]	10
4.1.1.2 Allgather() [2/2]	10
4.1.1.3 Allgatherv()	11
4.1.1.4 AllReduce() [1/2]	11
4.1.1.5 AllReduce() [2/2]	11
4.1.1.6 Barrier()	11
4.1.1.7 Bcast()	11
4.1.1.8 Bcast_managed()	12
4.1.1.9 Bcast_recv() [1/2]	12
4.1.1.10 Bcast_recv() [2/2]	12
4.1.1.11 Bcast_recv_managed()	12
4.1.1.12 Bcast_recv_one()	12
4.1.1.13 Bcast_send() [1/2]	13
4.1.1.14 Bcast_send() [2/2]	13
4.1.1.15 Bcast_send_managed() [1/2]	13
4.1.1.16 Bcast_send_managed() [2/2]	13
4.1.1.17 Bcast_send_one()	13
4.1.1.18 Comm_rank()	14
4.1.1.19 Comm_size()	14
4.1.1.20 Gather()	14
4.1.1.21 Gather_recv() [1/2]	14
4.1.1.22 Gather_recv() [2/2]	14
4.1.1.23 Gather_recv_one()	15
4.1.1.24 Gather_send() [1/2]	15
4.1.1.25 Gather_send() [2/2]	15
4.1.1.26 Gather_send_one()	15
4.1.1.27 Gatherv()	15
4.1.1.28 Gatherv_recv()	16
4.1.1.29 Gatherv_send()	16
4.1.1.30 Get_count() [1/2]	16
4.1.1.31 Get_count() [2/2]	16

4.1.1.32	Get_processor_name()	16
4.1.1.33	Group_rank()	17
4.1.1.34	Group_size()	17
4.1.1.35	Recv() [1/2]	17
4.1.1.36	Recv() [2/2]	17
4.1.1.37	Recv_one()	17
4.1.1.38	Reduce()	18
4.1.1.39	Reduce_recv() [1/2]	18
4.1.1.40	Reduce_recv() [2/2]	18
4.1.1.41	Reduce_send() [1/2]	18
4.1.1.42	Reduce_send() [2/2]	19
4.1.1.43	Scatter()	19
4.1.1.44	Scatter_recv() [1/2]	19
4.1.1.45	Scatter_recv() [2/2]	19
4.1.1.46	Scatter_recv_managed() [1/2]	20
4.1.1.47	Scatter_recv_managed() [2/2]	20
4.1.1.48	Scatter_send() [1/2]	20
4.1.1.49	Scatter_send() [2/2]	20
4.1.1.50	Scatter_send_managed() [1/2]	20
4.1.1.51	Scatter_send_managed() [2/2]	21
4.1.1.52	Scatterv()	21
4.1.1.53	Scatterv_recv()	21
4.1.1.54	Scatterv_send()	21
4.1.1.55	Send() [1/2]	21
4.1.1.56	Send() [2/2]	22
4.1.1.57	Send_one()	22
4.1.1.58	Type_size()	22
4.2	MPLw::details Namespace Reference	22
4.2.1	Function Documentation	22
4.2.1.1	split_buffer()	23
4.3	MPLw::details::cnpts Namespace Reference	23
4.3.1	Variable Documentation	23
4.3.1.1	Container	23
4.3.1.2	EnumOrInt	23
4.4	MPLw::errors Namespace Reference	23
4.4.1	Function Documentation	24
4.4.1.1	check_code()	24
4.4.1.2	error_message()	24
4.5	MPLw::structs Namespace Reference	24
4.6	MPLw::types Namespace Reference	24
4.6.1	Function Documentation	24
4.6.1.1	get_mpi_type()	24

<b>5 Class Documentation</b>	<b>25</b>
5.1 MPIw::Comm_raii Class Reference	25
5.1.1 Constructor & Destructor Documentation	25
5.1.1.1 Comm_raii() [1/3]	25
5.1.1.2 Comm_raii() [2/3]	26
5.1.1.3 Comm_raii() [3/3]	26
5.1.1.4 ~Comm_raii()	26
5.1.2 Member Function Documentation	26
5.1.2.1 get()	26
5.1.2.2 operator MPI_Comm()	26
5.1.2.3 operator&()	26
5.1.2.4 operator=() [1/2]	26
5.1.2.5 operator=() [2/2]	27
5.1.3 Member Data Documentation	27
5.1.3.1 comm	27
5.2 MPIw::Group_raii Class Reference	27
5.2.1 Constructor & Destructor Documentation	27
5.2.1.1 Group_raii() [1/3]	27
5.2.1.2 Group_raii() [2/3]	28
5.2.1.3 Group_raii() [3/3]	28
5.2.1.4 ~Group_raii()	28
5.2.2 Member Function Documentation	28
5.2.2.1 get()	28
5.2.2.2 operator MPI_Group()	28
5.2.2.3 operator&()	28
5.2.2.4 operator=() [1/2]	28
5.2.2.5 operator=() [2/2]	29
5.2.3 Member Data Documentation	29
5.2.3.1 group	29
5.3 MPIw::Init_raii Class Reference	29
5.3.1 Constructor & Destructor Documentation	29
5.3.1.1 Init_raii() [1/3]	29
5.3.1.2 Init_raii() [2/3]	30
5.3.1.3 Init_raii() [3/3]	30
5.3.1.4 ~Init_raii()	30
5.3.2 Member Function Documentation	30
5.3.2.1 operator=() [1/2]	30
5.3.2.2 operator=() [2/2]	30
5.4 MPIw::Init_threads_raii Class Reference	30
5.4.1 Constructor & Destructor Documentation	31
5.4.1.1 Init_threads_raii() [1/3]	31
5.4.1.2 Init_threads_raii() [2/3]	31

5.4.1.3 Init_threads_raii() [3/3] . . . . .	31
5.4.1.4 ~Init_threads_raii() . . . . .	31
5.4.2 Member Function Documentation . . . . .	31
5.4.2.1 operator=() [1/2] . . . . .	32
5.4.2.2 operator=() [2/2] . . . . .	32
5.4.2.3 support_level() . . . . .	32
5.5 MPIw::structs::Recv_st< T > Struct Template Reference . . . . .	32
5.5.1 Member Data Documentation . . . . .	32
5.5.1.1 data . . . . .	32
5.5.1.2 status . . . . .	33
5.6 MPIw::Type_raii Class Reference . . . . .	33
5.6.1 Constructor & Destructor Documentation . . . . .	33
5.6.1.1 Type_raii() [1/3] . . . . .	33
5.6.1.2 Type_raii() [2/3] . . . . .	33
5.6.1.3 Type_raii() [3/3] . . . . .	34
5.6.1.4 ~Type_raii() . . . . .	34
5.6.2 Member Function Documentation . . . . .	34
5.6.2.1 get() . . . . .	34
5.6.2.2 operator MPI_Datatype() . . . . .	34
5.6.2.3 operator&() . . . . .	34
5.6.2.4 operator=() [1/2] . . . . .	34
5.6.2.5 operator=() [2/2] . . . . .	34
5.6.3 Member Data Documentation . . . . .	34
5.6.3.1 type . . . . .	34
<b>6 File Documentation</b> . . . . .	<b>35</b>
6.1 /home/somik/Workspace/cpp/mpi_wrapper/src/communication.hpp File Reference . . . . .	35
6.2 /home/somik/Workspace/cpp/mpi_wrapper/src/concepts.hpp File Reference . . . . .	39
6.3 /home/somik/Workspace/cpp/mpi_wrapper/src/error_codes.hpp File Reference . . . . .	40
6.4 /home/somik/Workspace/cpp/mpi_wrapper/src/getters.hpp File Reference . . . . .	42
6.5 /home/somik/Workspace/cpp/mpi_wrapper/src/include.hpp File Reference . . . . .	44
6.6 /home/somik/Workspace/cpp/mpi_wrapper/src/raii.hpp File Reference . . . . .	44
6.7 /home/somik/Workspace/cpp/mpi_wrapper/src/structs.hpp File Reference . . . . .	45
6.8 /home/somik/Workspace/cpp/mpi_wrapper/src/types.hpp File Reference . . . . .	46
6.8.1 Macro Definition Documentation . . . . .	48
6.8.1.1 MPIw_register_type . . . . .	48
6.8.2 Function Documentation . . . . .	48
6.8.2.1 MPIw_register_type() [1/18] . . . . .	49
6.8.2.2 MPIw_register_type() [2/18] . . . . .	49
6.8.2.3 MPIw_register_type() [3/18] . . . . .	49
6.8.2.4 MPIw_register_type() [4/18] . . . . .	49
6.8.2.5 MPIw_register_type() [5/18] . . . . .	49

---

6.8.2.6 MPIw_register_type() [6/18] . . . . .	49
6.8.2.7 MPIw_register_type() [7/18] . . . . .	50
6.8.2.8 MPIw_register_type() [8/18] . . . . .	50
6.8.2.9 MPIw_register_type() [9/18] . . . . .	50
6.8.2.10 MPIw_register_type() [10/18] . . . . .	50
6.8.2.11 MPIw_register_type() [11/18] . . . . .	50
6.8.2.12 MPIw_register_type() [12/18] . . . . .	50
6.8.2.13 MPIw_register_type() [13/18] . . . . .	51
6.8.2.14 MPIw_register_type() [14/18] . . . . .	51
6.8.2.15 MPIw_register_type() [15/18] . . . . .	51
6.8.2.16 MPIw_register_type() [16/18] . . . . .	51
6.8.2.17 MPIw_register_type() [17/18] . . . . .	51
6.8.2.18 MPIw_register_type() [18/18] . . . . .	51
<b>Index</b>	<b>53</b>





# Chapter 1

## Namespace Index

### 1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">MPIw</a>	7
<a href="#">MPIw::details</a>	22
<a href="#">MPIw::details::cnpts</a>	23
<a href="#">MPIw::errors</a>	23
<a href="#">MPIw::structs</a>	24
<a href="#">MPIw::types</a>	24



## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">MPIw::Comm_raii</a>	25
<a href="#">MPIw::Group_raii</a>	27
<a href="#">MPIw::Init_raii</a>	29
<a href="#">MPIw::Init_threads_raii</a>	30
<a href="#">MPIw::structs::Recv_st&lt; T &gt;</a>	32
<a href="#">MPIw::Type_raii</a>	33



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

/home/somik/Workspace/cpp/mpi_wrapper/src/ <a href="#">communication.hpp</a> . . . . .	35
/home/somik/Workspace/cpp/mpi_wrapper/src/ <a href="#">concepts.hpp</a> . . . . .	39
/home/somik/Workspace/cpp/mpi_wrapper/src/ <a href="#">error_codes.hpp</a> . . . . .	40
/home/somik/Workspace/cpp/mpi_wrapper/src/ <a href="#">getters.hpp</a> . . . . .	42
/home/somik/Workspace/cpp/mpi_wrapper/src/ <a href="#">include.hpp</a> . . . . .	44
/home/somik/Workspace/cpp/mpi_wrapper/src/ <a href="#">raii.hpp</a> . . . . .	44
/home/somik/Workspace/cpp/mpi_wrapper/src/ <a href="#">structs.hpp</a> . . . . .	45
/home/somik/Workspace/cpp/mpi_wrapper/src/ <a href="#">types.hpp</a> . . . . .	46



## Chapter 4

# Namespace Documentation

### 4.1 MPIw Namespace Reference

#### Namespaces

- [details](#)
- [errors](#)
- [structs](#)
- [types](#)

#### Classes

- class [Init\\_raii](#)
- class [Init\\_threads\\_raii](#)
- class [Comm\\_raii](#)
- class [Group\\_raii](#)
- class [Type\\_raii](#)

#### Functions

- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`MPI_Status Recv (MPI_Comm comm, T *dest, int count, int source=MPI_ANY_SOURCE, U tag=MPI_↔ANY_TAG, const std::source_location &location=std::source_location::current())`
- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`structs::Recv\_st< std::vector< T > > Recv (MPI_Comm comm, int source=MPI_ANY_SOURCE, U tag=MPI_ANY_TAG, const std::source_location &location=std::source_location::current())`
- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`structs::Recv\_st< T > Recv\_one (MPI_Comm comm, int source=MPI_ANY_SOURCE, U tag=MPI_ANY_↔TAG, const std::source_location &location=std::source_location::current())`
- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`void Send (MPI_Comm comm, const T *data, int count, int dest, U tag, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T, details::cnpts::EnumOrInt U = int>`  
`void Send (MPI_Comm comm, const T &data, int dest, U tag, const std::source_location &location=std_↔::source_location::current())`

- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`void Send\_one (MPI_Comm comm, T data, int dest, U tag, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Bcast (MPI_Comm comm, const T &data, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Bcast\_managed (MPI_Comm comm, const T &data, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Bcast\_send (MPI_Comm comm, const T *data, int count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void Bcast\_send (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Bcast\_send\_one (MPI_Comm comm, T data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Bcast\_recv (MPI_Comm comm, T *dest, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > Bcast\_recv (MPI_Comm comm, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`T Bcast\_recv\_one (MPI_Comm comm, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Bcast\_send\_managed (MPI_Comm comm, const T *data, int count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void Bcast\_send\_managed (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > Bcast\_recv\_managed (MPI_Comm comm, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Gather (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Gather\_send (MPI_Comm comm, const T *data, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void Gather\_send (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Gather\_send\_one (MPI_Comm comm, T data, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Gather\_recv (MPI_Comm comm, const T *data, T *dest, int count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Gather\_recv (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > Gather\_recv\_one (MPI_Comm comm, T data, const std::source_location &location=std::source_location::current())`



- `template<typename T>`  
`void Allgather (MPI_Comm comm, const T *data, T *dest, int count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Allgather (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< std::vector< typename T::value_type > > Gather (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void Gather_send (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< std::vector< typename T::value_type > > Gather_recv (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< std::vector< typename T::value_type > > Allgather (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Scatter (MPI_Comm comm, const T &data, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Scatter_send (MPI_Comm comm, const T *data, T *dest, int total_count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Scatter_send (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Scatter_recv (MPI_Comm comm, T *dest, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > Scatter_recv (MPI_Comm comm, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Scatter_send_managed (MPI_Comm comm, const T *data, T *dest, int total_count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Scatter_send_managed (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void Scatter_recv_managed (MPI_Comm comm, T *dest, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > Scatter_recv_managed (MPI_Comm comm, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Scatter (MPI_Comm comm, const std::vector< T > &data, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Scatter_send (MPI_Comm comm, const std::vector< T > &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > Scatter_recv (MPI_Comm comm, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Reduce (MPI_Comm comm, const T &data, MPI_Op op, int root, const std::source_location &location=std::source_location::current())`

- `template<typename T >`  
`void Reduce_send (MPI_Comm comm, const T *data, int count, MPI_Op op, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void Reduce_send (MPI_Comm comm, const T &data, MPI_Op op, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`void Reduce_recv (MPI_Comm comm, const T *data, T *dest, int count, MPI_Op op, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > Reduce_recv (MPI_Comm comm, const T &data, MPI_Op op, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`void AllReduce (MPI_Comm comm, const T *data, T *dest, int count, MPI_Op op, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > AllReduce (MPI_Comm comm, const T &data, MPI_Op op, const std::source_location &location=std::source_location::current())`
- `void Barrier (MPI_Comm comm, const std::source_location &location=std::source_location::current())`
- `int Get_count (const MPI_Status &status, MPI_Datatype type, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`int Get_count (const MPI_Status &status, const std::source_location &location=std::source_location::current())`
- `int Comm_rank (MPI_Comm comm, const std::source_location &location=std::source_location::current())`
- `int Comm_size (MPI_Comm comm, const std::source_location &location=std::source_location::current())`
- `int Group_rank (MPI_Group group, const std::source_location &location=std::source_location::current())`
- `int Group_size (MPI_Group group, const std::source_location &location=std::source_location::current())`
- `std::string Get_processor_name (const std::source_location &location=std::source_location::current())`
- `int Type_size (MPI_Datatype type, const std::source_location &location=std::source_location::current())`

## 4.1.1 Function Documentation

### 4.1.1.1 Allgather() [1/2]

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Allgather (
    MPI_Comm comm,
    const T & data,
    const std::source_location & location = std::source_location::current() )
```

### 4.1.1.2 Allgather() [2/2]

```
template<typename T >
void MPIw::Allgather (
    MPI_Comm comm,
    const T * data,
    T * dest,
    int count,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.3 Allgatherv()

```
template<details::cnpts::Container T>
std::vector<std::vector<typename T::value_type> > MPIw::Allgatherv (
    MPI_Comm comm,
    const T & data,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.4 AllReduce() [1/2]

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::AllReduce (
    MPI_Comm comm,
    const T & data,
    MPI_Op op,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.5 AllReduce() [2/2]

```
template<typename T >
void MPIw::AllReduce (
    MPI_Comm comm,
    const T * data,
    T * dest,
    int count,
    MPI_Op op,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.6 Barrier()

```
void MPIw::Barrier (
    MPI_Comm comm,
    const std::source_location & location = std::source_location::current() ) [inline]
```

#### 4.1.1.7 Bcast()

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Bcast (
    MPI_Comm comm,
    const T & data,
    int count,
    int root,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.8 Bcast\_managed()

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Bcast_managed (
    MPI_Comm comm,
    const T & data,
    int count,
    int root,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.9 Bcast\_recv() [1/2]

```
template<typename T >
std::vector<T> MPIw::Bcast_recv (
    MPI_Comm comm,
    int count,
    int root,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.10 Bcast\_recv() [2/2]

```
template<typename T >
void MPIw::Bcast_recv (
    MPI_Comm comm,
    T * dest,
    int count,
    int root,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.11 Bcast\_recv\_managed()

```
template<typename T >
std::vector<T> MPIw::Bcast_recv_managed (
    MPI_Comm comm,
    int root,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.12 Bcast\_recv\_one()

```
template<typename T >
T MPIw::Bcast_recv_one (
    MPI_Comm comm,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.13 Bcast\_send() [1/2]**

```
template<details::cnpts::Container T>
void MPIw::Bcast_send (
    MPI_Comm comm,
    const T & data,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.14 Bcast\_send() [2/2]**

```
template<typename T >
void MPIw::Bcast_send (
    MPI_Comm comm,
    const T * data,
    int count,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.15 Bcast\_send\_managed() [1/2]**

```
template<details::cnpts::Container T>
void MPIw::Bcast_send_managed (
    MPI_Comm comm,
    const T & data,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.16 Bcast\_send\_managed() [2/2]**

```
template<typename T >
void MPIw::Bcast_send_managed (
    MPI_Comm comm,
    const T * data,
    int count,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.17 Bcast\_send\_one()**

```
template<typename T >
void MPIw::Bcast_send_one (
    MPI_Comm comm,
    T data,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.18 Comm\_rank()**

```
int MPIw::Comm_rank (
    MPI_Comm comm,
    const std::source_location & location = std::source_location::current() ) [inline]
```

**4.1.1.19 Comm\_size()**

```
int MPIw::Comm_size (
    MPI_Comm comm,
    const std::source_location & location = std::source_location::current() ) [inline]
```

**4.1.1.20 Gather()**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Gather (
    MPI_Comm comm,
    const T & data,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.21 Gather\_recv() [1/2]**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Gather_recv (
    MPI_Comm comm,
    const T & data,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.22 Gather\_recv() [2/2]**

```
template<typename T >
void MPIw::Gather_recv (
    MPI_Comm comm,
    const T * data,
    T * dest,
    int count,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.23 Gather\_recv\_one()

```
template<typename T >
std::vector<T> MPIw::Gather_recv_one (
    MPI_Comm comm,
    T data,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.24 Gather\_send() [1/2]

```
template<details::cnpts::Container T>
void MPIw::Gather_send (
    MPI_Comm comm,
    const T & data,
    int root,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.25 Gather\_send() [2/2]

```
template<typename T >
void MPIw::Gather_send (
    MPI_Comm comm,
    const T * data,
    int count,
    int root,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.26 Gather\_send\_one()

```
template<typename T >
void MPIw::Gather_send_one (
    MPI_Comm comm,
    T data,
    int root,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.27 Gatherv()

```
template<details::cnpts::Container T>
std::vector<std::vector<typename T::value_type> > > MPIw::Gatherv (
    MPI_Comm comm,
    const T & data,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.28 Gatherv\_recv()**

```
template<details::cnpts::Container T>
std::vector<std::vector<typename T::value_type> > MPIw::Gatherv_recv (
    MPI_Comm comm,
    const T & data,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.29 Gatherv\_send()**

```
template<details::cnpts::Container T>
void MPIw::Gatherv_send (
    MPI_Comm comm,
    const T & data,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.30 Get\_count() [1/2]**

```
template<typename T >
int MPIw::Get_count (
    const MPI_Status & status,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.31 Get\_count() [2/2]**

```
int MPIw::Get_count (
    const MPI_Status & status,
    MPI_Datatype type,
    const std::source_location & location = std::source_location::current() ) [inline]
```

**4.1.1.32 Get\_processor\_name()**

```
std::string MPIw::Get_processor_name (
    const std::source_location & location = std::source_location::current() ) [inline]
```



#### 4.1.1.33 Group\_rank()

```
int MPIw::Group_rank (
    MPI_Group group,
    const std::source_location & location = std::source_location::current() ) [inline]
```

#### 4.1.1.34 Group\_size()

```
int MPIw::Group_size (
    MPI_Group group,
    const std::source_location & location = std::source_location::current() ) [inline]
```

#### 4.1.1.35 Recv() [1/2]

```
template<typename T , details::cnpts::EnumOrInt U = int>
structs::Recv_st<std::vector<T> > MPIw::Recv (
    MPI_Comm comm,
    int source = MPI_ANY_SOURCE,
    U tag = MPI_ANY_TAG,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.36 Recv() [2/2]

```
template<typename T , details::cnpts::EnumOrInt U = int>
MPI_Status MPIw::Recv (
    MPI_Comm comm,
    T * dest,
    int count,
    int source = MPI_ANY_SOURCE,
    U tag = MPI_ANY_TAG,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.37 Recv\_one()

```
template<typename T , details::cnpts::EnumOrInt U = int>
structs::Recv_st<T> MPIw::Recv_one (
    MPI_Comm comm,
    int source = MPI_ANY_SOURCE,
    U tag = MPI_ANY_TAG,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.38 Reduce()**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Reduce (
    MPI_Comm comm,
    const T & data,
    MPI_Op op,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.39 Reduce\_recv() [1/2]**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Reduce_recv (
    MPI_Comm comm,
    const T & data,
    MPI_Op op,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.40 Reduce\_recv() [2/2]**

```
template<typename T >
void MPIw::Reduce_recv (
    MPI_Comm comm,
    const T * data,
    T * dest,
    int count,
    MPI_Op op,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.41 Reduce\_send() [1/2]**

```
template<details::cnpts::Container T>
void MPIw::Reduce_send (
    MPI_Comm comm,
    const T & data,
    MPI_Op op,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.42 Reduce\_send() [2/2]**

```
template<typename T >
void MPIw::Reduce_send (
    MPI_Comm comm,
    const T * data,
    int count,
    MPI_Op op,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.43 Scatter()**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Scatter (
    MPI_Comm comm,
    const T & data,
    int count,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.44 Scatter\_recv() [1/2]**

```
template<typename T >
std::vector<T> MPIw::Scatter_recv (
    MPI_Comm comm,
    int count,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.45 Scatter\_recv() [2/2]**

```
template<typename T >
void MPIw::Scatter_recv (
    MPI_Comm comm,
    T * dest,
    int count,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.46 Scatter\_recv\_managed() [1/2]**

```
template<typename T >
std::vector<T> MPIw::Scatter_recv_managed (
    MPI_Comm comm,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.47 Scatter\_recv\_managed() [2/2]**

```
template<typename T >
void MPIw::Scatter_recv_managed (
    MPI_Comm comm,
    T * dest,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.48 Scatter\_send() [1/2]**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Scatter_send (
    MPI_Comm comm,
    const T & data,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.49 Scatter\_send() [2/2]**

```
template<typename T >
void MPIw::Scatter_send (
    MPI_Comm comm,
    const T * data,
    T * dest,
    int total_count,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.50 Scatter\_send\_managed() [1/2]**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Scatter_send_managed (
    MPI_Comm comm,
    const T & data,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.51 Scatter\_send\_managed()** [2/2]

```
template<typename T >
void MPIw::Scatter_send_managed (
    MPI_Comm comm,
    const T * data,
    T * dest,
    int total_count,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.52 Scatterv()**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Scatterv (
    MPI_Comm comm,
    const std::vector< T > & data,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.53 Scatterv\_recv()**

```
template<typename T >
std::vector<T> MPIw::Scatterv_recv (
    MPI_Comm comm,
    int root,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.54 Scatterv\_send()**

```
template<details::cnpts::Container T>
std::vector<typename T::value_type> MPIw::Scatterv_send (
    MPI_Comm comm,
    const std::vector< T > & data,
    const std::source_location & location = std::source_location::current() )
```

**4.1.1.55 Send()** [1/2]

```
template<details::cnpts::Container T, details::cnpts::EnumOrInt U = int>
void MPIw::Send (
    MPI_Comm comm,
    const T & data,
    int dest,
    U tag,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.56 Send() [2/2]

```
template<typename T , details::cnpts::EnumOrInt U = int>
void MPIw::Send (
    MPI_Comm comm,
    const T * data,
    int count,
    int dest,
    U tag,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.57 Send\_one()

```
template<typename T , details::cnpts::EnumOrInt U = int>
void MPIw::Send_one (
    MPI_Comm comm,
    T data,
    int dest,
    U tag,
    const std::source_location & location = std::source_location::current() )
```

#### 4.1.1.58 Type\_size()

```
int MPIw::Type_size (
    MPI_Datatype type,
    const std::source_location & location = std::source_location::current() ) [inline]
```

## 4.2 MPIw::details Namespace Reference

### Namespaces

- [cnpts](#)

### Functions

- `template<typename T >`  
`std::vector< std::vector< T > > split\_buffer (const std::vector< T > &buffer, const std::vector< int > &offsets)`

#### 4.2.1 Function Documentation

#### 4.2.1.1 split\_buffer()

```
template<typename T >
std::vector<std::vector<T> > MPIw::details::split_buffer (
    const std::vector< T > & buffer,
    const std::vector< int > & offsets )
```

## 4.3 MPIw::details::cnpts Namespace Reference

### Variables

- template<typename T >  
concept [EnumOrInt](#)
- template<typename T >  
concept [Container](#)

### 4.3.1 Variable Documentation

#### 4.3.1.1 Container

```
template<typename T >
concept MPIw::details::cnpts::Container
```

##### Initial value:

```
= requires(T a) {
    { begin(a) } -> std::contiguous_iterator;
    { a.size() } -> std::same_as<std::size_t>;
    { sizeof(typename T::value_type) } -> std::same_as<std::size_t>;
}
```

#### 4.3.1.2 EnumOrInt

```
template<typename T >
concept MPIw::details::cnpts::EnumOrInt
```

##### Initial value:

```
= requires(T) {
    requires std::is_enum_v<T> || std::is_same_v<T, int>;
}
```

## 4.4 MPIw::errors Namespace Reference

### Functions

- std::string [error\\_message](#) (int error\_code)
- void [check\\_code](#) (int error\_code, const std::source\_location &location=std::source\_location::current())

### 4.4.1 Function Documentation

#### 4.4.1.1 `check_code()`

```
void MPIw::errors::check_code (
    int error_code,
    const std::source_location & location = std::source_location::current() ) [inline]
```

#### 4.4.1.2 `error_message()`

```
std::string MPIw::errors::error_message (
    int error_code ) [inline]
```

## 4.5 MPIw::structs Namespace Reference

### Classes

- struct [Recv\\_st](#)

## 4.6 MPIw::types Namespace Reference

### Functions

- template<typename T >  
MPI\_Datatype [get\\_mpi\\_type](#) (T=T{})

### 4.6.1 Function Documentation

#### 4.6.1.1 `get_mpi_type()`

```
template<typename T >
MPI_Datatype MPIw::types::get_mpi_type (
    T = T{} )
```



## Chapter 5

# Class Documentation

### 5.1 MPIw::Comm\_raii Class Reference

```
#include <raii.hpp>
```

#### Public Member Functions

- [Comm\\_raii](#) ()=default
- [Comm\\_raii](#) (const [Comm\\_raii](#) &)=delete
- [Comm\\_raii](#) & [operator=](#) (const [Comm\\_raii](#) &)=delete
- [Comm\\_raii](#) ([Comm\\_raii](#) &&)=delete
- [Comm\\_raii](#) && [operator=](#) ([Comm\\_raii](#) &&)=delete
- [~Comm\\_raii](#) ()
- MPI\_Comm & [get](#) ()
- [operator MPI\\_Comm](#) ()
- MPI\_Comm \* [operator&](#) ()

#### Public Attributes

- MPI\_Comm [comm](#) = MPI\_COMM\_NULL

#### 5.1.1 Constructor & Destructor Documentation

##### 5.1.1.1 Comm\_raii() [1/3]

```
MPIw::Comm_raii::Comm_raii ( ) [default]
```

#### 5.1.1.2 Comm\_raii() [2/3]

```
MPIw::Comm_raii::Comm_raii (
    const Comm_raii & ) [delete]
```

#### 5.1.1.3 Comm\_raii() [3/3]

```
MPIw::Comm_raii::Comm_raii (
    Comm_raii && ) [delete]
```

#### 5.1.1.4 ~Comm\_raii()

```
MPIw::Comm_raii::~~Comm_raii ( ) [inline]
```

### 5.1.2 Member Function Documentation

#### 5.1.2.1 get()

```
MPI_Comm& MPIw::Comm_raii::get ( ) [inline]
```

#### 5.1.2.2 operator MPI\_Comm()

```
MPIw::Comm_raii::operator MPI_Comm ( ) [inline]
```

#### 5.1.2.3 operator&()

```
MPI_Comm* MPIw::Comm_raii::operator& ( ) [inline]
```

#### 5.1.2.4 operator=() [1/2]

```
Comm_raii& MPIw::Comm_raii::operator= (
    Comm_raii && ) [delete]
```

### 5.1.2.5 operator=() [2/2]

```
Comm_raii& MPIw::Comm_raii::operator= (
    const Comm_raii & ) [delete]
```

## 5.1.3 Member Data Documentation

### 5.1.3.1 comm

```
MPI_Comm MPIw::Comm_raii::comm = MPI_COMM_NULL
```

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

- [/home/somik/Workspace/cpp/mpi\\_wrapper/src/raii.hpp](#)

## 5.2 MPIw::Group\_raii Class Reference

```
#include <raii.hpp>
```

### Public Member Functions

- [Group\\_raii](#) ()=default
- [Group\\_raii](#) (const [Group\\_raii](#) &)=delete
- [Group\\_raii](#) & [operator=](#) (const [Group\\_raii](#) &)=delete
- [Group\\_raii](#) ([Group\\_raii](#) &&)=delete
- [Group\\_raii](#) && [operator=](#) ([Group\\_raii](#) &&)=delete
- [~Group\\_raii](#) ()
- MPI\_Group & [get](#) ()
- [operator MPI\\_Group](#) ()
- MPI\_Group \* [operator&](#) ()

### Public Attributes

- MPI\_Group [group](#) = MPI\_GROUP\_NULL

## 5.2.1 Constructor & Destructor Documentation

### 5.2.1.1 Group\_raii() [1/3]

```
MPIw::Group_raii::Group_raii ( ) [default]
```

#### 5.2.1.2 Group\_raii() [2/3]

```
MPIw::Group_raii::Group_raii (
    const Group_raii & ) [delete]
```

#### 5.2.1.3 Group\_raii() [3/3]

```
MPIw::Group_raii::Group_raii (
    Group_raii && ) [delete]
```

#### 5.2.1.4 ~Group\_raii()

```
MPIw::Group_raii::~~Group_raii ( ) [inline]
```

### 5.2.2 Member Function Documentation

#### 5.2.2.1 get()

```
MPI_Group& MPIw::Group_raii::get ( ) [inline]
```

#### 5.2.2.2 operator MPI\_Group()

```
MPIw::Group_raii::operator MPI_Group ( ) [inline]
```

#### 5.2.2.3 operator&()

```
MPI_Group* MPIw::Group_raii::operator& ( ) [inline]
```

#### 5.2.2.4 operator=() [1/2]

```
Group_raii& MPIw::Group_raii::operator= (
    const Group_raii & ) [delete]
```

### 5.2.2.5 operator=() [2/2]

```
Group_raii&& MPIw::Group_raii::operator= (
    Group_raii && ) [delete]
```

## 5.2.3 Member Data Documentation

### 5.2.3.1 group

```
MPI_Group MPIw::Group_raii::group = MPI_GROUP_NULL
```

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

- [/home/somik/Workspace/cpp/mpi\\_wrapper/src/raii.hpp](/home/somik/Workspace/cpp/mpi_wrapper/src/raii.hpp)

## 5.3 MPIw::Init\_raii Class Reference

```
#include <raii.hpp>
```

### Public Member Functions

- [Init\\_raii](#) (int \*argc, char \*\*\*argv)
- [Init\\_raii](#) (const [Init\\_raii](#) &)=delete
- [Init\\_raii](#) & [operator=](#) (const [Init\\_raii](#) &)=delete
- [Init\\_raii](#) ([Init\\_raii](#) &&)=delete
- [Init\\_raii](#) && [operator=](#) ([Init\\_raii](#) &&)=delete
- [~Init\\_raii](#) ()

### 5.3.1 Constructor & Destructor Documentation

#### 5.3.1.1 Init\_raii() [1/3]

```
MPIw::Init_raii::Init_raii (
    int * argc,
    char *** argv ) [inline]
```

### 5.3.1.2 Init\_raii() [2/3]

```
MPIw::Init_raii::Init_raii (
    const Init_raii & ) [delete]
```

### 5.3.1.3 Init\_raii() [3/3]

```
MPIw::Init_raii::Init_raii (
    Init_raii && ) [delete]
```

### 5.3.1.4 ~Init\_raii()

```
MPIw::Init_raii::~~Init_raii ( ) [inline]
```

## 5.3.2 Member Function Documentation

### 5.3.2.1 operator=() [1/2]

```
Init_raii& MPIw::Init_raii::operator= (
    const Init_raii & ) [delete]
```

### 5.3.2.2 operator=() [2/2]

```
Init_raii&& MPIw::Init_raii::operator= (
    Init_raii && ) [delete]
```

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

- /home/somik/Workspace/cpp/mpi\_wrapper/src/[raii.hpp](#)

## 5.4 MPIw::Init\_threads\_raii Class Reference

```
#include <raii.hpp>
```

## Public Member Functions

- [Init\\_threads\\_raii](#) (int \*argc, char \*\*\*argv, int required)
- [Init\\_threads\\_raii](#) (const [Init\\_threads\\_raii](#) &)=delete
- [Init\\_threads\\_raii](#) & [operator=](#) (const [Init\\_threads\\_raii](#) &)=delete
- [Init\\_threads\\_raii](#) ([Init\\_threads\\_raii](#) &&)=delete
- [Init\\_threads\\_raii](#) && [operator=](#) ([Init\\_threads\\_raii](#) &&)=delete
- [~Init\\_threads\\_raii](#) ()
- [int support\\_level](#) () const

## 5.4.1 Constructor & Destructor Documentation

### 5.4.1.1 Init\_threads\_raii() [1/3]

```
MPIw::Init_threads_raii::Init_threads_raii (
    int * argc,
    char *** argv,
    int required ) [inline]
```

### 5.4.1.2 Init\_threads\_raii() [2/3]

```
MPIw::Init_threads_raii::Init_threads_raii (
    const Init\_threads\_raii & ) [delete]
```

### 5.4.1.3 Init\_threads\_raii() [3/3]

```
MPIw::Init_threads_raii::Init_threads_raii (
    Init\_threads\_raii && ) [delete]
```

### 5.4.1.4 ~Init\_threads\_raii()

```
MPIw::Init_threads_raii::~~Init_threads_raii ( ) [inline]
```

## 5.4.2 Member Function Documentation

#### 5.4.2.1 operator=() [1/2]

```
Init_threads_raii& MPIw::Init_threads_raii::operator= (
    const Init_threads_raii & ) [delete]
```

#### 5.4.2.2 operator=() [2/2]

```
Init_threads_raii&& MPIw::Init_threads_raii::operator= (
    Init_threads_raii && ) [delete]
```

#### 5.4.2.3 support\_level()

```
int MPIw::Init_threads_raii::support_level ( ) const [inline]
```

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

- [/home/somik/Workspace/cpp/mpi\\_wrapper/src/raii.hpp](#)

## 5.5 MPIw::structs::Recv\_st< T > Struct Template Reference

```
#include <structs.hpp>
```

### Public Attributes

- T [data](#)
- MPI\_Status [status](#)

### 5.5.1 Member Data Documentation

#### 5.5.1.1 data

```
template<typename T >
T MPIw::structs::Recv_st< T >::data
```



### 5.5.1.2 status

```
template<typename T >
MPI_Status MPIw::structs::Recv_st< T >::status
```

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

- /home/somik/Workspace/cpp/mpi\_wrapper/src/structs.hpp

## 5.6 MPIw::Type\_raii Class Reference

```
#include <raii.hpp>
```

### Public Member Functions

- [Type\\_raii](#) ()=default
- [Type\\_raii](#) (const [Type\\_raii](#) &)=delete
- [Type\\_raii](#) & [operator=](#) (const [Type\\_raii](#) &)=delete
- [Type\\_raii](#) ([Type\\_raii](#) &&)=delete
- [Type\\_raii](#) && [operator=](#) ([Type\\_raii](#) &&)=delete
- [~Type\\_raii](#) ()
- MPI\_Datatype & [get](#) ()
- [operator MPI\\_Datatype](#) ()
- MPI\_Datatype \* [operator&](#) ()

### Public Attributes

- MPI\_Datatype [type](#) = MPI\_DATATYPE\_NULL

## 5.6.1 Constructor & Destructor Documentation

### 5.6.1.1 [Type\\_raii](#)() [1/3]

```
MPIw::Type_raii::Type_raii ( ) [default]
```

### 5.6.1.2 [Type\\_raii](#)() [2/3]

```
MPIw::Type_raii::Type_raii (
    const Type\_raii & ) [delete]
```

### 5.6.1.3 Type\_raii() [3/3]

```
MPIw::Type_raii::Type_raii (
    Type_raii && ) [delete]
```

### 5.6.1.4 ~Type\_raii()

```
MPIw::Type_raii::~~Type_raii ( ) [inline]
```

## 5.6.2 Member Function Documentation

### 5.6.2.1 get()

```
MPI_Datatype& MPIw::Type_raii::get ( ) [inline]
```

### 5.6.2.2 operator MPI\_Datatype()

```
MPIw::Type_raii::operator MPI_Datatype ( ) [inline]
```

### 5.6.2.3 operator&()

```
MPI_Datatype* MPIw::Type_raii::operator& ( ) [inline]
```

### 5.6.2.4 operator=() [1/2]

```
Type_raii& MPIw::Type_raii::operator= (
    const Type_raii & ) [delete]
```

### 5.6.2.5 operator=() [2/2]

```
Type_raii&& MPIw::Type_raii::operator= (
    Type_raii && ) [delete]
```

## 5.6.3 Member Data Documentation

### 5.6.3.1 type

```
MPI_Datatype MPIw::Type_raii::type = MPI_DATATYPE_NULL
```

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

- [/home/somik/Workspace/cpp/mpi\\_wrapper/src/raii.hpp](/home/somik/Workspace/cpp/mpi_wrapper/src/raii.hpp)

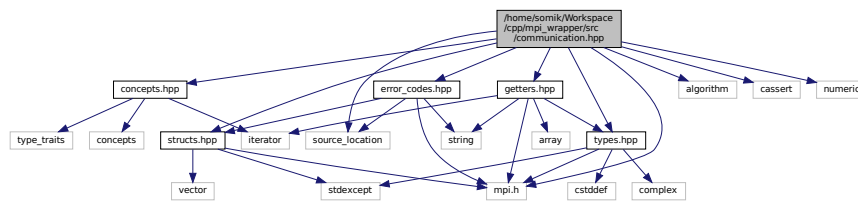
## Chapter 6

# File Documentation

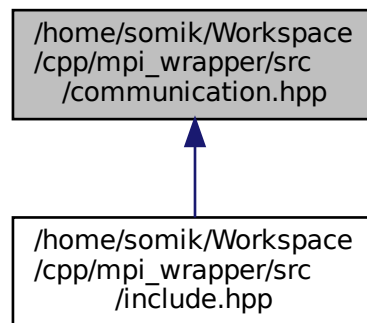
### 6.1 /home/somik/Workspace/cpp/mpi\_wrapper/src/communication.hpp File Reference

```
#include "concepts.hpp"  
#include "error_codes.hpp"  
#include "getters.hpp"  
#include "structs.hpp"  
#include "types.hpp"  
#include <algorithm>  
#include <cassert>  
#include <mpi.h>  
#include <numeric>  
#include <source_location>
```

Include dependency graph for communication.hpp:



This graph shows which files directly or indirectly include this file:



## Namespaces

- [MPIw](#)
- [MPIw::details](#)

## Functions

- `template<typename T >`  
`std::vector< std::vector< T > > MPIw::details::split\_buffer (const std::vector< T > &buffer, const std::vector< int > &offsets)`
- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`MPI_Status MPIw::Recv (MPI_Comm comm, T *dest, int count, int source=MPI_ANY_SOURCE, U tag=MPI_ANY_TAG, const std::source_location &location=std::source_location::current())`
- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`structs::Recv_st< std::vector< T > > MPIw::Recv (MPI_Comm comm, int source=MPI_ANY_SOURCE, U tag=MPI_ANY_TAG, const std::source_location &location=std::source_location::current())`
- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`structs::Recv_st< T > MPIw::Recv\_one (MPI_Comm comm, int source=MPI_ANY_SOURCE, U tag=MPI_ANY_TAG, const std::source_location &location=std::source_location::current())`
- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`void MPIw::Send (MPI_Comm comm, const T *data, int count, int dest, U tag, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T, details::cnpts::EnumOrInt U = int>`  
`void MPIw::Send (MPI_Comm comm, const T &data, int dest, U tag, const std::source_location &location=std::source_location::current())`
- `template<typename T, details::cnpts::EnumOrInt U = int>`  
`void MPIw::Send\_one (MPI_Comm comm, T data, int dest, U tag, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Bcast (MPI_Comm comm, const T &data, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Bcast\_managed (MPI_Comm comm, const T &data, int count, int root, const std::source_location &location=std::source_location::current())`

- `template<typename T>`  
`void MPIw::Bcast_send (MPI_Comm comm, const T *data, int count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void MPIw::Bcast_send (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void MPIw::Bcast_send_one (MPI_Comm comm, T data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void MPIw::Bcast_recv (MPI_Comm comm, T *dest, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > MPIw::Bcast_recv (MPI_Comm comm, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`T MPIw::Bcast_recv_one (MPI_Comm comm, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void MPIw::Bcast_send_managed (MPI_Comm comm, const T *data, int count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void MPIw::Bcast_send_managed (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > MPIw::Bcast_recv_managed (MPI_Comm comm, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Gather (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void MPIw::Gather_send (MPI_Comm comm, const T *data, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void MPIw::Gather_send (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void MPIw::Gather_send_one (MPI_Comm comm, T data, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void MPIw::Gather_recv (MPI_Comm comm, const T *data, T *dest, int count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Gather_recv (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`std::vector< T > MPIw::Gather_recv_one (MPI_Comm comm, T data, const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void MPIw::Allgather (MPI_Comm comm, const T *data, T *dest, int count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Allgather (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< std::vector< typename T::value_type > > MPIw::Gatherv (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())`

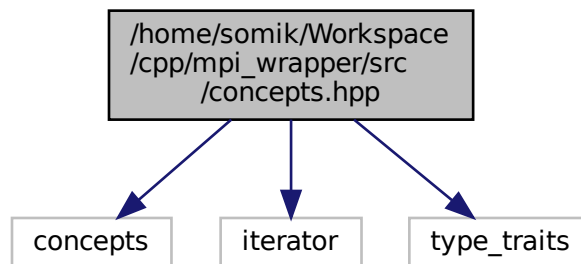
- `template<details::cnpts::Container T>`  
`void MPIw::Gatherv_send (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< std::vector< typename T::value_type > > MPIw::Gatherv_recv (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< std::vector< typename T::value_type > > MPIw::Allgatherv (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Scatter (MPI_Comm comm, const T &data, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`void MPIw::Scatter_send (MPI_Comm comm, const T *data, T *dest, int total_count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Scatter_send (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`void MPIw::Scatter_recv (MPI_Comm comm, T *dest, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`std::vector< T > MPIw::Scatter_recv (MPI_Comm comm, int count, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`void MPIw::Scatter_send_managed (MPI_Comm comm, const T *data, T *dest, int total_count, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Scatter_send_managed (MPI_Comm comm, const T &data, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`void MPIw::Scatter_recv_managed (MPI_Comm comm, T *dest, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`std::vector< T > MPIw::Scatter_recv_managed (MPI_Comm comm, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Scatterv (MPI_Comm comm, const std::vector< T > &data, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Scatterv_send (MPI_Comm comm, const std::vector< T > &data, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`std::vector< T > MPIw::Scatterv_recv (MPI_Comm comm, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Reduce (MPI_Comm comm, const T &data, MPI_Op op, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`void MPIw::Reduce_send (MPI_Comm comm, const T *data, int count, MPI_Op op, int root, const std::source_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`void MPIw::Reduce_send (MPI_Comm comm, const T &data, MPI_Op op, int root, const std::source_location &location=std::source_location::current())`
- `template<typename T >`  
`void MPIw::Reduce_recv (MPI_Comm comm, const T *data, T *dest, int count, MPI_Op op, const std::source_location &location=std::source_location::current())`

- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::Reduce_recv (MPI_Comm comm, const T &data, MPI_Op op,`  
`const std::source_location &location=std::source_location::current())`
- `template<typename T>`  
`void MPIw::AllReduce (MPI_Comm comm, const T *data, T *dest, int count, MPI_Op op, const std::source_↵`  
`_location &location=std::source_location::current())`
- `template<details::cnpts::Container T>`  
`std::vector< typename T::value_type > MPIw::AllReduce (MPI_Comm comm, const T &data, MPI_Op op,`  
`const std::source_location &location=std::source_location::current())`
- `void MPIw::Barrier (MPI_Comm comm, const std::source_location &location=std::source_location::current())`

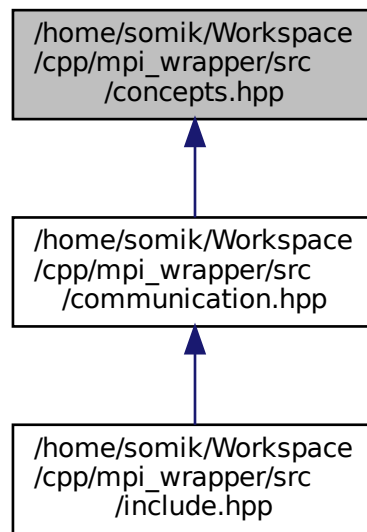
## 6.2 /home/somik/Workspace/cpp/mpi\_wrapper/src/concepts.hpp File Reference

```
#include <concepts>
#include <iterator>
#include <type_traits>
```

Include dependency graph for concepts.hpp:



This graph shows which files directly or indirectly include this file:



## Namespaces

- [MPIw](#)
- [MPIw::details](#)
- [MPIw::details::cnpts](#)

## Variables

- `template<typename T >`  
concept [MPIw::details::cnpts::EnumOrInt](#)
- `template<typename T >`  
concept [MPIw::details::cnpts::Container](#)

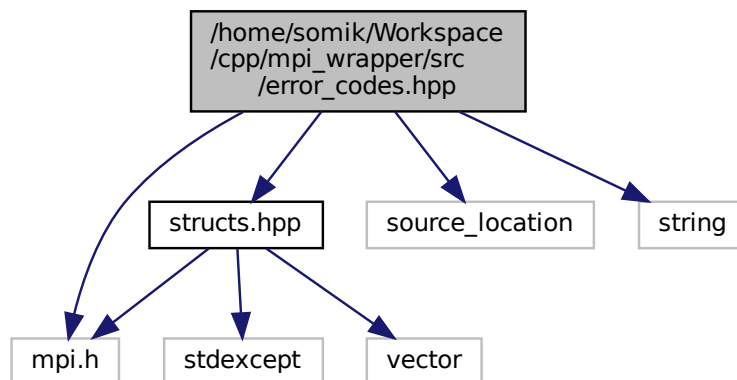
## 6.3 /home/somik/Workspace/cpp/mpi\_wrapper/src/error\_codes.hpp File Reference

```
#include "structs.hpp"
#include <mpi.h>
#include <source_location>
```

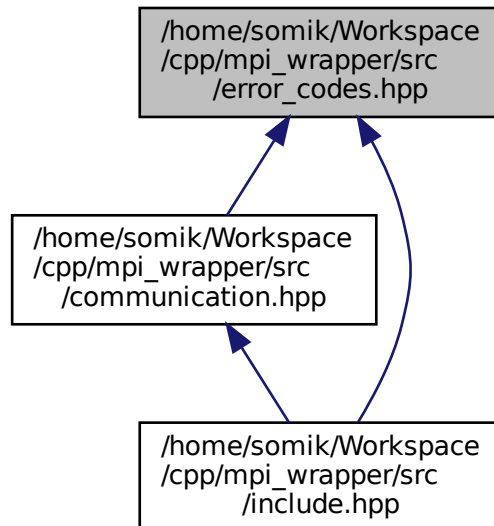


```
#include <string>
```

Include dependency graph for error\_codes.hpp:



This graph shows which files directly or indirectly include this file:



## Namespaces

- [MPIw](#)
- [MPIw::errors](#)

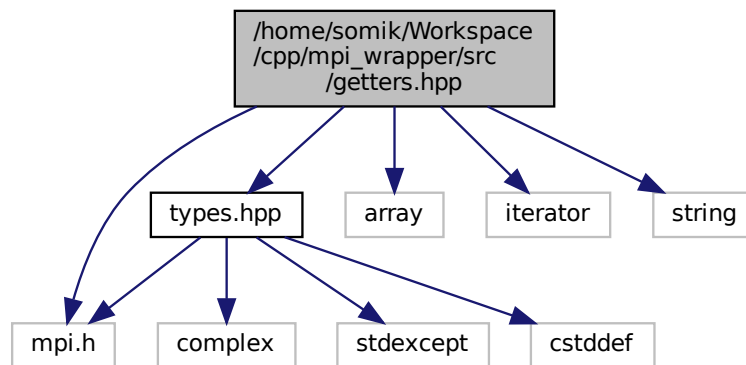
## Functions

- std::string [MPIw::errors::error\\_message](#) (int error\_code)
- void [MPIw::errors::check\\_code](#) (int error\_code, const std::source\_location &location=std::source\_location::current())

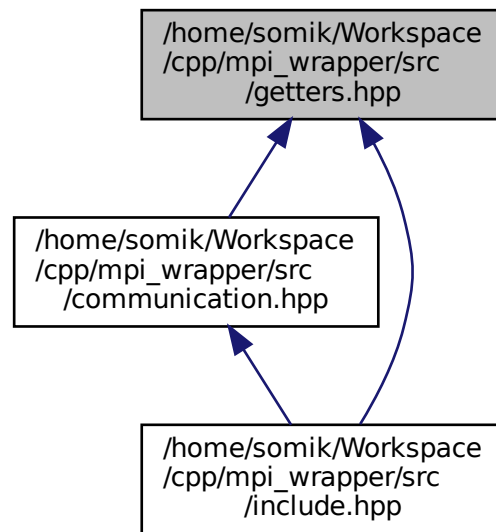
## 6.4 /home/somik/Workspace/cpp/mpi\_wrapper/src/getters.hpp File Reference

```
#include "types.hpp"
#include <array>
#include <iterator>
#include <mpi.h>
#include <string>
```

Include dependency graph for getters.hpp:



This graph shows which files directly or indirectly include this file:



## Namespaces

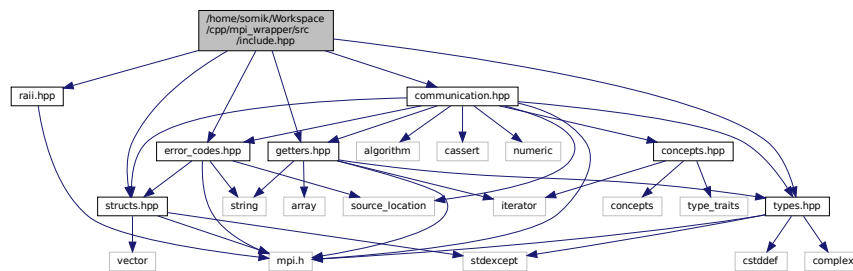
- [MPIw](#)

## Functions

- `int MPIw::Get\_count (const MPI_Status &status, MPI_Datatype type, const std::source_location &location=std::source_location::current())`
- `template<typename T >  
int MPIw::Get\_count (const MPI_Status &status, const std::source_location &location=std::source_location::current())`
- `int MPIw::Comm\_rank (MPI_Comm comm, const std::source_location &location=std::source_location::current())`
- `int MPIw::Comm\_size (MPI_Comm comm, const std::source_location &location=std::source_location::current())`
- `int MPIw::Group\_rank (MPI_Group group, const std::source_location &location=std::source_location::current())`
- `int MPIw::Group\_size (MPI_Group group, const std::source_location &location=std::source_location::current())`
- `std::string MPIw::Get\_processor\_name (const std::source_location &location=std::source_location::current())`
- `int MPIw::Type\_size (MPI_Datatype type, const std::source_location &location=std::source_location::current())`

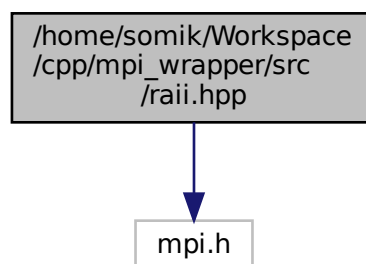
## 6.5 /home/somik/Workspace/cpp/mpi\_wrapper/src/include.hpp File Reference

```
#include "communication.hpp"
#include "getters.hpp"
#include "raii.hpp"
#include "structs.hpp"
#include "types.hpp"
#include "error_codes.hpp"
Include dependency graph for include.hpp:
```

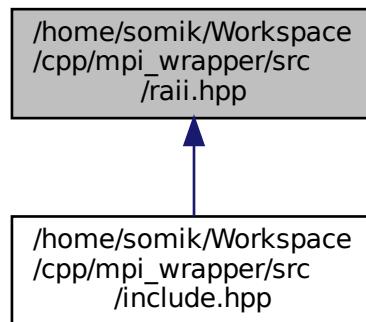


## 6.6 /home/somik/Workspace/cpp/mpi\_wrapper/src/raii.hpp File Reference

```
#include <mpi.h>
Include dependency graph for raii.hpp:
```



This graph shows which files directly or indirectly include this file:



### Classes

- class `MPIw::Init_raii`
- class `MPIw::Init_threads_raii`
- class `MPIw::Comm_raii`
- class `MPIw::Group_raii`
- class `MPIw::Type_raii`

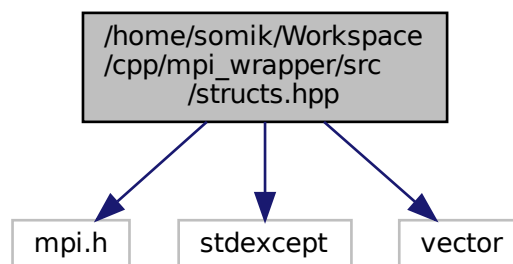
### Namespaces

- `MPIw`

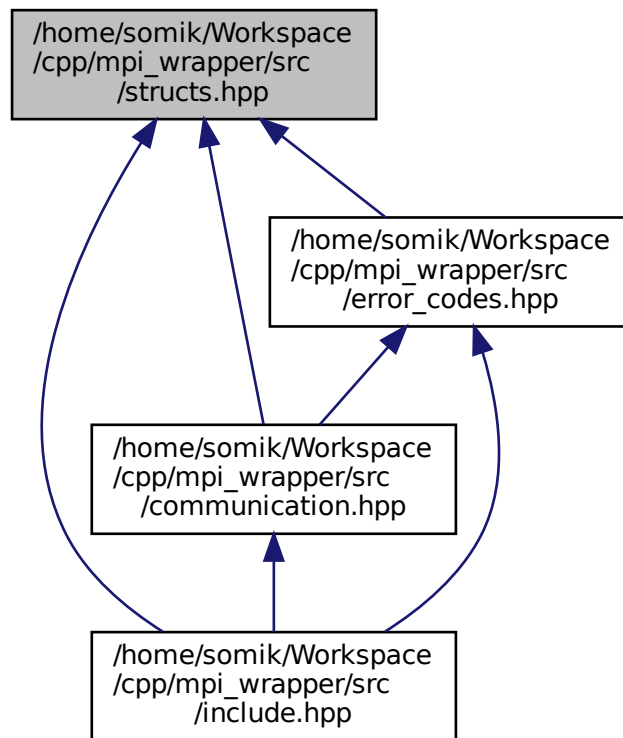
## 6.7 /home/somik/Workspace/cpp/mpi\_wrapper/src/structs.hpp File Reference

```
#include <mpi.h>
#include <stdexcept>
#include <vector>
```

Include dependency graph for structs.hpp:



This graph shows which files directly or indirectly include this file:



## Classes

- struct [MPIw::structs::Recv\\_st< T >](#)

## Namespaces

- [MPIw](#)
- [MPIw::structs](#)

## 6.8 /home/somik/Workspace/cpp/mpi\_wrapper/src/types.hpp File Reference

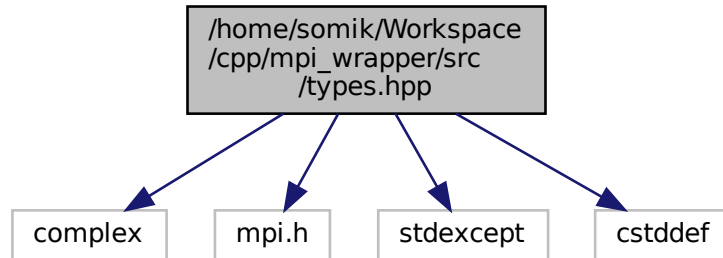
```

#include <complex>
#include <mpi.h>
#include <stdexcept>

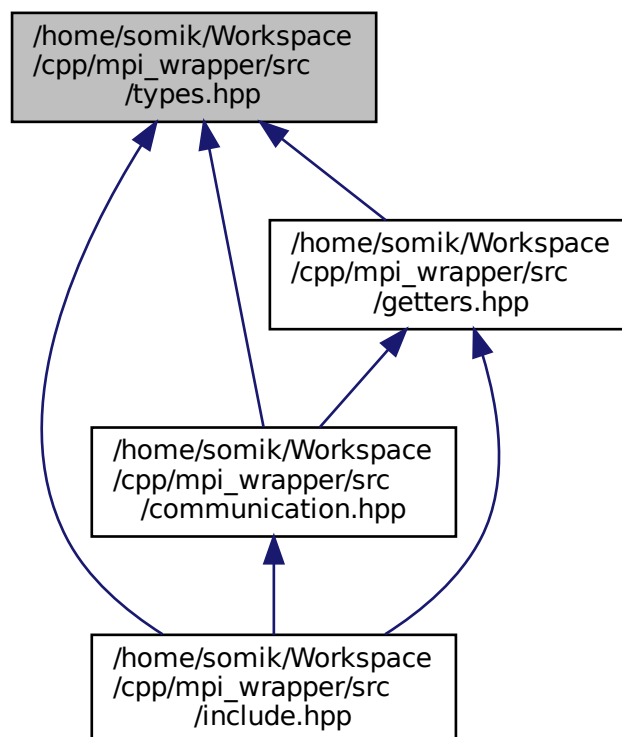
```

```
#include <cstdlib>
```

Include dependency graph for types.hpp:



This graph shows which files directly or indirectly include this file:



## Namespaces

- [MPIw](#)
- [MPIw::types](#)

## Macros

- `#define MPIw_register_type(cpp_type, mpi_type)`

## Functions

- `template<typename T >`  
`MPI_Datatype MPIw::types::get_mpi_type (T=T{})`
- `MPIw_register_type (std::byte, MPI_BYTE)`
- `MPIw_register_type (char, MPI_CHAR)`
- `MPIw_register_type (wchar_t, MPI_WCHAR)`
- `MPIw_register_type (short, MPI_SHORT)`
- `MPIw_register_type (int, MPI_INT)`
- `MPIw_register_type (long, MPI_LONG)`
- `MPIw_register_type (signed char, MPI_SIGNED_CHAR)`
- `MPIw_register_type (unsigned char, MPI_UNSIGNED_CHAR)`
- `MPIw_register_type (unsigned short, MPI_UNSIGNED_SHORT)`
- `MPIw_register_type (unsigned, MPI_UNSIGNED)`
- `MPIw_register_type (unsigned long, MPI_UNSIGNED_LONG)`
- `MPIw_register_type (float, MPI_FLOAT)`
- `MPIw_register_type (double, MPI_DOUBLE)`
- `MPIw_register_type (long double, MPI_LONG_DOUBLE)`
- `MPIw_register_type (bool, MPI_CXX_BOOL)`
- `MPIw_register_type (std::complex< float >, MPI_CXX_COMPLEX)`
- `MPIw_register_type (std::complex< double >, MPI_CXX_DOUBLE_COMPLEX)`
- `MPIw_register_type (std::complex< long double >, MPI_CXX_LONG_DOUBLE_COMPLEX)`

## 6.8.1 Macro Definition Documentation

### 6.8.1.1 MPIw\_register\_type

```
#define MPIw_register_type(  
    cpp_type,  
    mpi_type )
```

#### Value:

```
namespace MPIw::types {  
    template <>  
    inline MPI_Datatype get_mpi_type<cpp_type>(cpp_type) {  
        return mpi_type;  
    }  
}
```

```
\\  
\\  
\\  
\\
```

## 6.8.2 Function Documentation



### 6.8.2.1 MPIw\_register\_type() [1/18]

```
MPIw_register_type (  
    bool ,  
    MPI_CXX_BOOL )
```

### 6.8.2.2 MPIw\_register\_type() [2/18]

```
MPIw_register_type (  
    char ,  
    MPI_CHAR )
```

### 6.8.2.3 MPIw\_register\_type() [3/18]

```
MPIw_register_type (  
    double ,  
    MPI_DOUBLE )
```

### 6.8.2.4 MPIw\_register\_type() [4/18]

```
MPIw_register_type (  
    float ,  
    MPI_FLOAT )
```

### 6.8.2.5 MPIw\_register\_type() [5/18]

```
MPIw_register_type (  
    int ,  
    MPI_INT )
```

### 6.8.2.6 MPIw\_register\_type() [6/18]

```
MPIw_register_type (  
    long double ,  
    MPI_LONG_DOUBLE )
```

**6.8.2.7 MPIw\_register\_type() [7/18]**

```
MPIw_register_type (
    long ,
    MPI_LONG )
```

**6.8.2.8 MPIw\_register\_type() [8/18]**

```
MPIw_register_type (
    short ,
    MPI_SHORT )
```

**6.8.2.9 MPIw\_register\_type() [9/18]**

```
MPIw_register_type (
    signed char ,
    MPI_SIGNED_CHAR )
```

**6.8.2.10 MPIw\_register\_type() [10/18]**

```
MPIw_register_type (
    std::byte ,
    MPI_BYTE )
```

**6.8.2.11 MPIw\_register\_type() [11/18]**

```
MPIw_register_type (
    std::complex< double > ,
    MPI_CXX_DOUBLE_COMPLEX )
```

**6.8.2.12 MPIw\_register\_type() [12/18]**

```
MPIw_register_type (
    std::complex< float > ,
    MPI_CXX_COMPLEX )
```

**6.8.2.13 MPIw\_register\_type() [13/18]**

```
MPIw_register_type (
    std::complex< long double > ,
    MPI_CXX_LONG_DOUBLE_COMPLEX )
```

**6.8.2.14 MPIw\_register\_type() [14/18]**

```
MPIw_register_type (
    unsigned char ,
    MPI_UNSIGNED_CHAR )
```

**6.8.2.15 MPIw\_register\_type() [15/18]**

```
MPIw_register_type (
    unsigned long ,
    MPI_UNSIGNED_LONG )
```

**6.8.2.16 MPIw\_register\_type() [16/18]**

```
MPIw_register_type (
    unsigned short ,
    MPI_UNSIGNED_SHORT )
```

**6.8.2.17 MPIw\_register\_type() [17/18]**

```
MPIw_register_type (
    unsigned ,
    MPI_UNSIGNED )
```

**6.8.2.18 MPIw\_register\_type() [18/18]**

```
MPIw_register_type (
    wchar_t ,
    MPI_WCHAR )
```



# Index

[/home/somik/Workspace/cpp/mpi\\_wrapper/src/communication.hpp](#), 35

[/home/somik/Workspace/cpp/mpi\\_wrapper/src/concepts.hpp](#), 39

[/home/somik/Workspace/cpp/mpi\\_wrapper/src/error\\_codes.hpp](#), 40

[/home/somik/Workspace/cpp/mpi\\_wrapper/src/getters.hpp](#), 42

[/home/somik/Workspace/cpp/mpi\\_wrapper/src/include.hpp](#), 44

[/home/somik/Workspace/cpp/mpi\\_wrapper/src/raii.hpp](#), 44

[/home/somik/Workspace/cpp/mpi\\_wrapper/src/structs.hpp](#), 45

[/home/somik/Workspace/cpp/mpi\\_wrapper/src/types.hpp](#), 46

[~Comm\\_raii](#)  
MPIw::Comm\_raii, 26

[~Group\\_raii](#)  
MPIw::Group\_raii, 28

[~Init\\_raii](#)  
MPIw::Init\_raii, 30

[~Init\\_threads\\_raii](#)  
MPIw::Init\_threads\_raii, 31

[~Type\\_raii](#)  
MPIw::Type\_raii, 34

[Allgather](#)  
MPIw, 10

[Allgatherv](#)  
MPIw, 10

[AllReduce](#)  
MPIw, 11

[Barrier](#)  
MPIw, 11

[Bcast](#)  
MPIw, 11

[Bcast\\_managed](#)  
MPIw, 11

[Bcast\\_recv](#)  
MPIw, 12

[Bcast\\_recv\\_managed](#)  
MPIw, 12

[Bcast\\_recv\\_one](#)  
MPIw, 12

[Bcast\\_send](#)  
MPIw, 12, 13

[Bcast\\_send\\_managed](#)  
MPIw, 13

[Bcast\\_send\\_one](#)  
MPIw, 13

[check\\_code](#)  
MPIw::errors, 24

[comm](#)  
MPIw::Comm\_raii, 27

[Comm\\_raii](#)  
MPIw::Comm\_raii, 25, 26

[Comm\\_rank](#)  
MPIw, 13

[Comm\\_size](#)  
MPIw, 14

[Container](#)  
MPIw::details::cnpts, 23

[data](#)  
MPIw::structs::Recv\_st< T >, 32

[EnumOrInt](#)  
MPIw::details::cnpts, 23

[error\\_message](#)  
MPIw::errors, 24

[Gather](#)  
MPIw, 14

[Gather\\_recv](#)  
MPIw, 14

[Gather\\_recv\\_one](#)  
MPIw, 14

[Gather\\_send](#)  
MPIw, 15

[Gather\\_send\\_one](#)  
MPIw, 15

[Gatherv](#)  
MPIw, 15

[Gatherv\\_recv](#)  
MPIw, 15

[Gatherv\\_send](#)  
MPIw, 16

[get](#)  
MPIw::Comm\_raii, 26  
MPIw::Group\_raii, 28  
MPIw::Type\_raii, 34

[Get\\_count](#)  
MPIw, 16

[get\\_mpi\\_type](#)  
MPIw::types, 24

[Get\\_processor\\_name](#)  
MPIw, 16

- group
  - MPIw::Group\_raii, 29
- Group\_raii
  - MPIw::Group\_raii, 27, 28
- Group\_rank
  - MPIw, 16
- Group\_size
  - MPIw, 17
- Init\_raii
  - MPIw::Init\_raii, 29, 30
- Init\_threads\_raii
  - MPIw::Init\_threads\_raii, 31
- MPIw, 7
  - Allgather, 10
  - Allgatherv, 10
  - AllReduce, 11
  - Barrier, 11
  - Bcast, 11
  - Bcast\_managed, 11
  - Bcast\_recv, 12
  - Bcast\_recv\_managed, 12
  - Bcast\_recv\_one, 12
  - Bcast\_send, 12, 13
  - Bcast\_send\_managed, 13
  - Bcast\_send\_one, 13
  - Comm\_rank, 13
  - Comm\_size, 14
  - Gather, 14
  - Gather\_recv, 14
  - Gather\_recv\_one, 14
  - Gather\_send, 15
  - Gather\_send\_one, 15
  - Gatherv, 15
  - Gatherv\_recv, 15
  - Gatherv\_send, 16
  - Get\_count, 16
  - Get\_processor\_name, 16
  - Group\_rank, 16
  - Group\_size, 17
  - Recv, 17
  - Recv\_one, 17
  - Reduce, 17
  - Reduce\_recv, 18
  - Reduce\_send, 18
  - Scatter, 19
  - Scatter\_recv, 19
  - Scatter\_recv\_managed, 19, 20
  - Scatter\_send, 20
  - Scatter\_send\_managed, 20
  - Scatterv, 21
  - Scatterv\_recv, 21
  - Scatterv\_send, 21
  - Send, 21
  - Send\_one, 22
  - Type\_size, 22
- MPIw::Comm\_raii, 25
  - ~Comm\_raii, 26
- comm, 27
- Comm\_raii, 25, 26
- get, 26
- operator MPI\_Comm, 26
- operator=, 26
- operator&, 26
- MPIw::details, 22
  - split\_buffer, 22
- MPIw::details::cnpts, 23
  - Container, 23
  - EnumOrInt, 23
- MPIw::errors, 23
  - check\_code, 24
  - error\_message, 24
- MPIw::Group\_raii, 27
  - ~Group\_raii, 28
  - get, 28
  - group, 29
  - Group\_raii, 27, 28
  - operator MPI\_Group, 28
  - operator=, 28
  - operator&, 28
- MPIw::Init\_raii, 29
  - ~Init\_raii, 30
  - Init\_raii, 29, 30
  - operator=, 30
- MPIw::Init\_threads\_raii, 30
  - ~Init\_threads\_raii, 31
  - Init\_threads\_raii, 31
  - operator=, 31, 32
  - support\_level, 32
- MPIw::structs, 24
- MPIw::structs::Recv\_st< T >, 32
  - data, 32
  - status, 32
- MPIw::Type\_raii, 33
  - ~Type\_raii, 34
  - get, 34
  - operator MPI\_Datatype, 34
  - operator=, 34
  - operator&, 34
  - type, 34
  - Type\_raii, 33
- MPIw::types, 24
  - get\_mpi\_type, 24
- MPIw\_register\_type
  - types.hpp, 48–51
- operator MPI\_Comm
  - MPIw::Comm\_raii, 26
- operator MPI\_Datatype
  - MPIw::Type\_raii, 34
- operator MPI\_Group
  - MPIw::Group\_raii, 28
- operator=
  - MPIw::Comm\_raii, 26
  - MPIw::Group\_raii, 28
  - MPIw::Init\_raii, 30
  - MPIw::Init\_threads\_raii, 31, 32

- MPIw::Type\_raii, [34](#)
- operator&
  - MPIw::Comm\_raii, [26](#)
  - MPIw::Group\_raii, [28](#)
  - MPIw::Type\_raii, [34](#)
- Recv
  - MPIw, [17](#)
- Recv\_one
  - MPIw, [17](#)
- Reduce
  - MPIw, [17](#)
- Reduce\_recv
  - MPIw, [18](#)
- Reduce\_send
  - MPIw, [18](#)
- Scatter
  - MPIw, [19](#)
- Scatter\_recv
  - MPIw, [19](#)
- Scatter\_recv\_managed
  - MPIw, [19](#), [20](#)
- Scatter\_send
  - MPIw, [20](#)
- Scatter\_send\_managed
  - MPIw, [20](#)
- Scatterv
  - MPIw, [21](#)
- Scatterv\_recv
  - MPIw, [21](#)
- Scatterv\_send
  - MPIw, [21](#)
- Send
  - MPIw, [21](#)
- Send\_one
  - MPIw, [22](#)
- split\_buffer
  - MPIw::details, [22](#)
- status
  - MPIw::structs::Recv\_st< T >, [32](#)
- support\_level
  - MPIw::Init\_threads\_raii, [32](#)
- type
  - MPIw::Type\_raii, [34](#)
- Type\_raii
  - MPIw::Type\_raii, [33](#)
- Type\_size
  - MPIw, [22](#)
- types.hpp
  - MPIw\_register\_type, [48–51](#)