MPI Wrapper

Generated by Doxygen 1.9.1

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

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

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

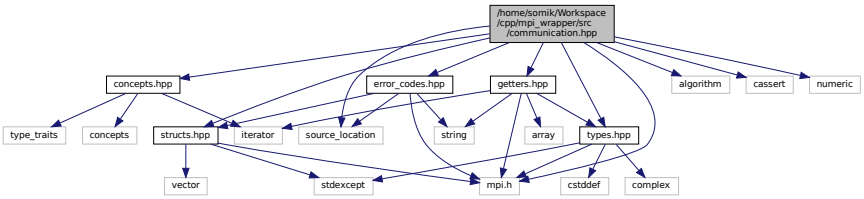
Chapter 6

File Documentation

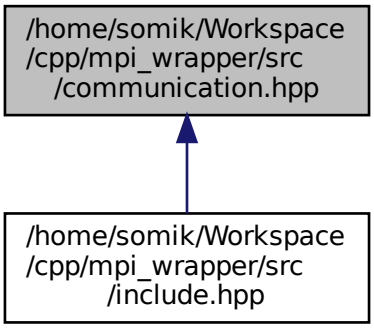
6.1 /home/somik/Workspace/cpp/mpi_wrapper/src/communication.hpp File Reference

```
#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())

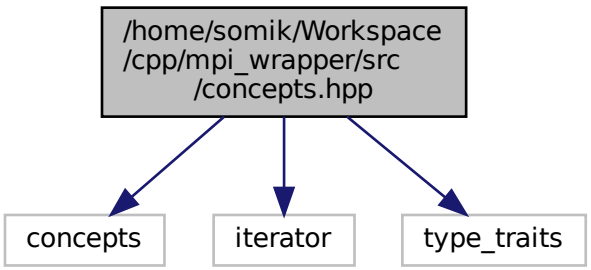
- template<details::cnpts::Container T>
  void MPIw::Gatherv_send (MPI_Comm comm, const T &data, int root, const std::source_location &location=std::source_location::current())
- template<details::cnpts::Container T>
  std::vector< std::vector< typename T::value_type > > MPIw::Gatherv_recv (MPI_Comm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

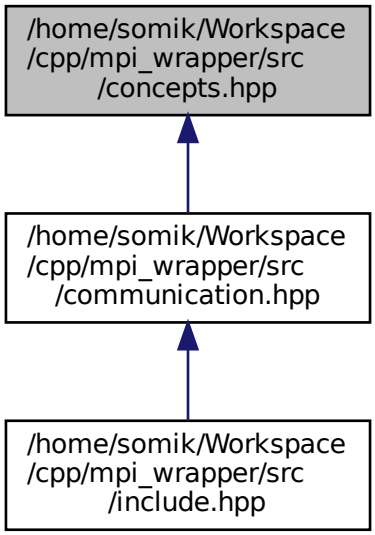
6.2 /home/somik/Workspace/cpp/mpi_wrapper/src/concepts.hpp File Reference

```
#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

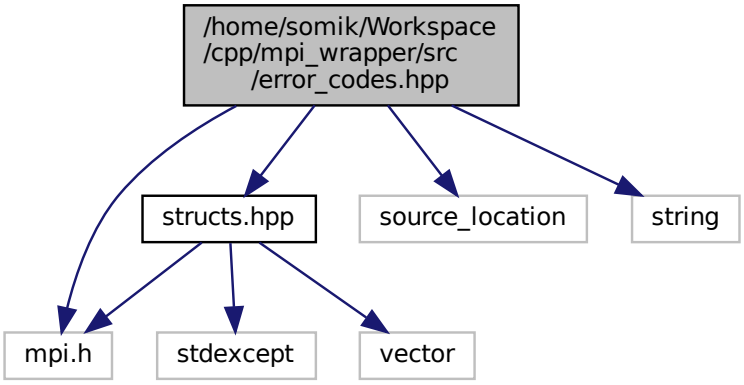
- template<typename T >
  concept MPIw::details::cnpts::EnumOrInt
- template<typename T >
  concept MPIw::details::cnpts::Container

6.3 /home/somik/Workspace/cpp/mpi_wrapper/src/error_codes.hpp File Reference

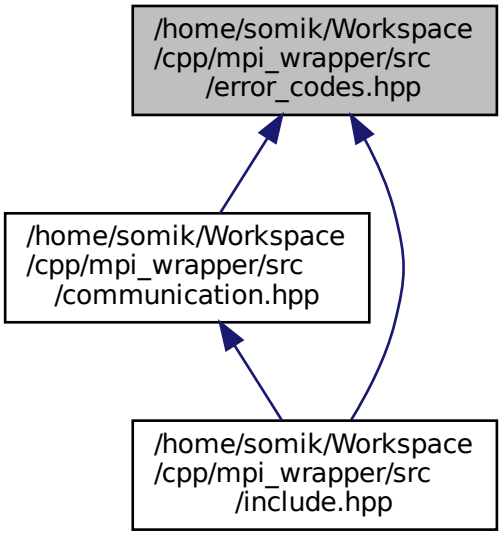
```
#include "structs.hpp"
#include <mpi.h>
#include <source_location>
```

```
#include <string>
```

Include dependency graph for error_codes.hpp:



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



Namespaces

- MPIw
- MPIw::errors

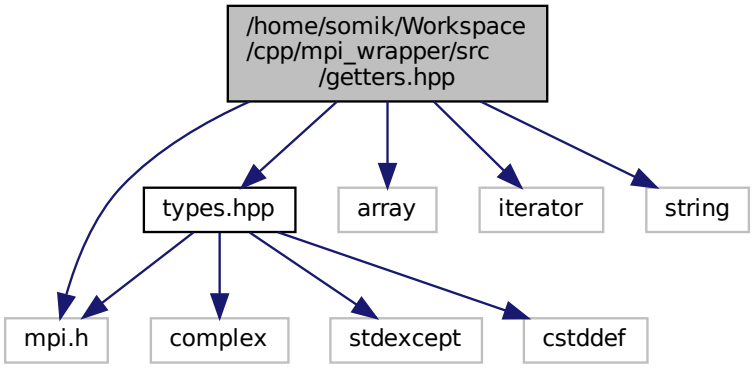
Functions

- 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())

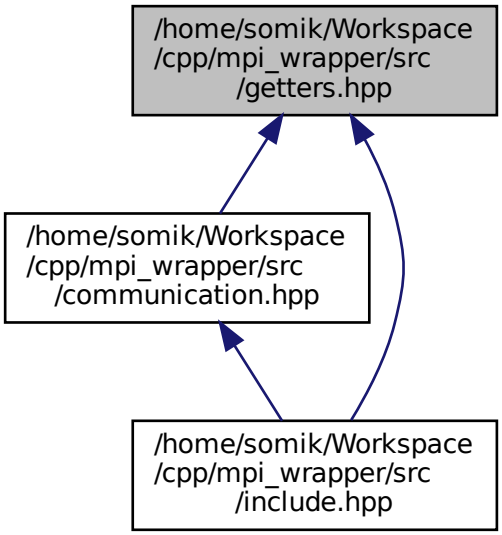
6.4 /home/somik/Workspace/cpp/mpi_wrapper/src/getters.hpp File Reference

```
#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

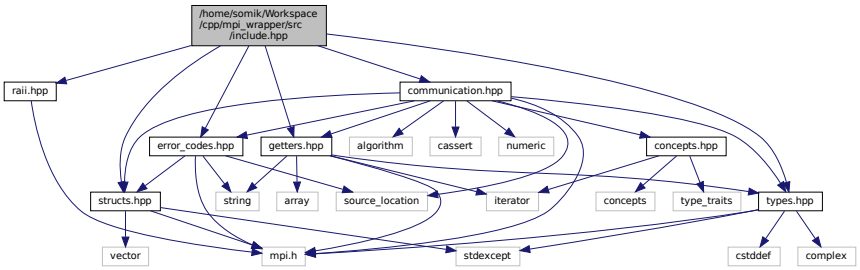
Functions

- 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"
```

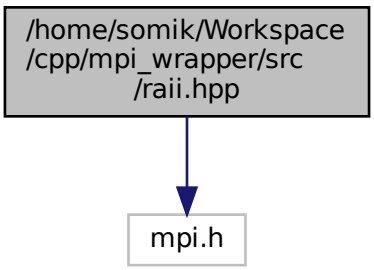
Include dependency graph for include.hpp:



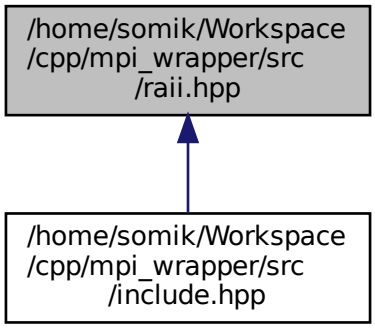
6.6 /home/somik/Workspace/cpp/mpi_wrapper/src/raii.hpp File Reference

```
#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

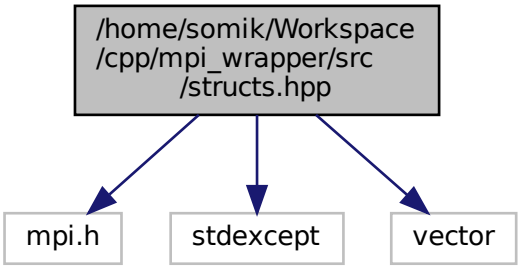
Namespaces

- MPIw

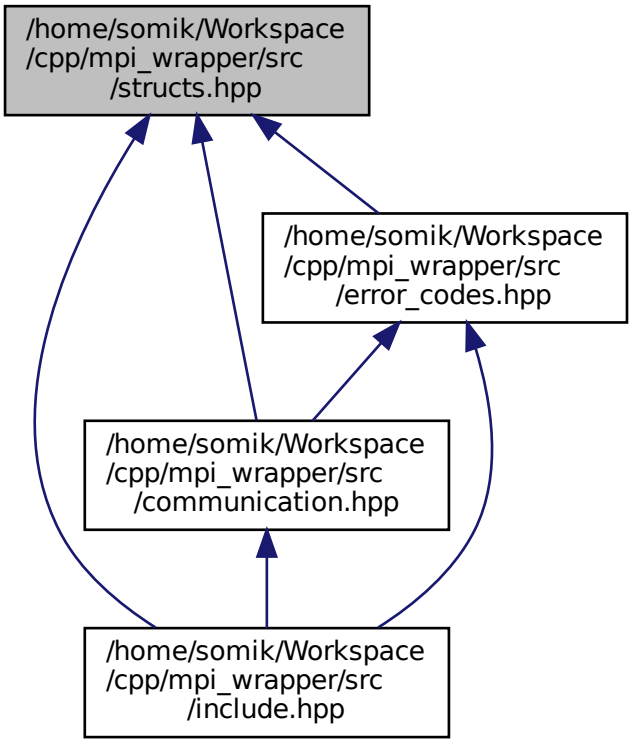
6.7 /home/somik/Workspace/cpp/mpi_wrapper/src/structs.hpp File Reference

```
#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

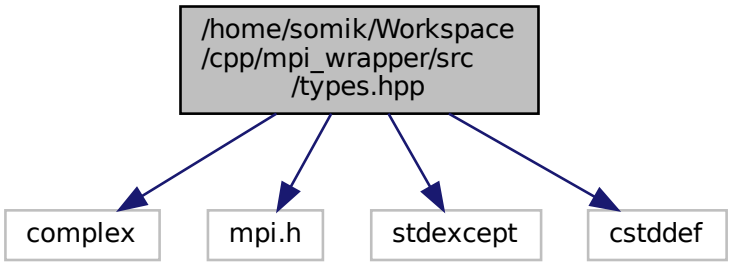
- MPIw
- MPIw::structs

6.8 /home/somik/Workspace/cpp/mpi_wrapper/src/types.hpp File Reference

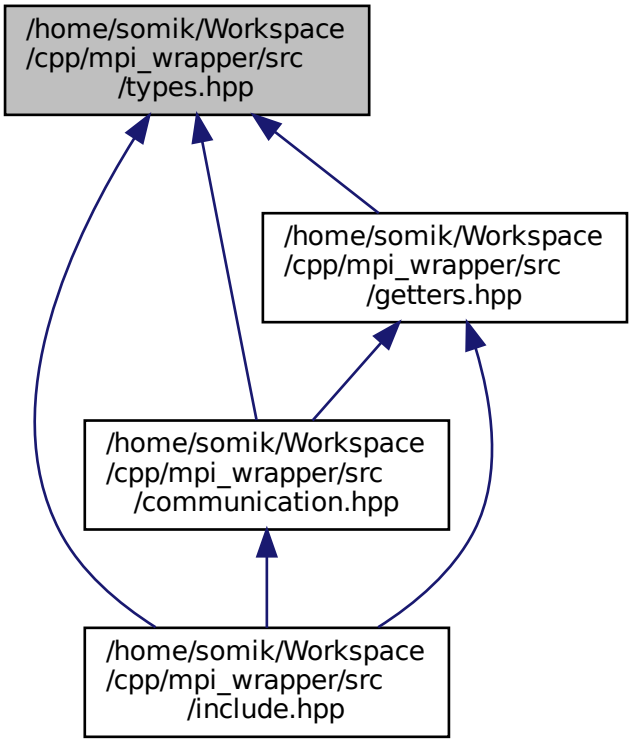
```
#include <complex>
#include <mpi.h>
#include <stdexcept>
```

```
#include <cstddef>
```

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