330. Defined Constants

Up: Defined Values and Handles **Next:** Types **Previous:** Defined Values and Handles

The C and Fortran name is listed in the left column and the C++ name is listed in the middle or right column. Constants with the type `const int` may also be implemented as literal integer constants substituted by the preprocessor.

Return Codes

C type: `const int` (or unnamed `enum`) Fortran type: `INTEGER`	C++ type: `const int` (or unnamed `enum`)
MPI_SUCCESS	MPI::SUCCESS
MPI_ERR_BUFFER	MPI::ERR_BUFFER
MPI_ERR_COUNT	MPI::ERR_COUNT
MPI_ERR_TYPE	MPI::ERR_TYPE
MPI_ERR_TAG	MPI::ERR_TAG
MPI_ERR_COMM	MPI::ERR_COMM
MPI_ERR_RANK	MPI::ERR_RANK
MPI_ERR_REQUEST	MPI::ERR_REQUEST
MPI_ERR_ROOT	MPI::ERR_ROOT
MPI_ERR_GROUP	MPI::ERR_GROUP
MPI_ERR_OP	MPI::ERR_OP
MPI_ERR_TOPOLOGY	MPI::ERR_TOPOLOGY
MPI_ERR_DIMS	MPI::ERR_DIMS
MPI_ERR_ARG	MPI::ERR_ARG
MPI_ERR_UNKNOWN	MPI::ERR_UNKNOWN
MPI_ERR_TRUNCATE	MPI::ERR_TRUNCATE
MPI_ERR_OTHER	MPI::ERR_OTHER
MPI_ERR_INTERN	MPI::ERR_INTERN
MPI_ERR_PENDING	MPI::ERR_PENDING

(Continued on next page)

Return Codes (continued)

MPI_ERR_IN_STATUS	MPI::ERR_IN_STATUS
MPI_ERR_ACCESS	MPI::ERR_ACCESS
MPI_ERR_AMODE	MPI::ERR_AMODE
MPI_ERR_ASSERT	MPI::ERR_ASSERT
MPI_ERR_BAD_FILE	MPI::ERR_BAD_FILE
MPI_ERR_BASE	MPI::ERR_BASE
MPI_ERR_CONVERSION	MPI::ERR_CONVERSION
MPI_ERR_DISP	MPI::ERR_DISP
MPI_ERR_DUP_DATAREP	MPI::ERR_DUP_DATAREP
MPI_ERR_FILE_EXISTS	MPI::ERR_FILE_EXISTS
MPI_ERR_FILE_IN_USE	MPI::ERR_FILE_IN_USE
MPI_ERR_FILE	MPI::ERR_FILE
MPI_ERR_INFO_KEY	MPI::ERR_INFO_VALUE

MPI_ERR_INFO_NOKEY	MPI::ERR_INFO_NOKEY
MPI_ERR_INFO_VALUE	MPI::ERR_INFO_KEY
MPI_ERR_INFO	MPI::ERR_INFO
MPI_ERR_IO	MPI::ERR_IO
MPI_ERR_KEYVAL	MPI::ERR_KEYVAL
MPI_ERR_LOCKTYPE	MPI::ERR_LOCKTYPE
MPI_ERR_NAME	MPI::ERR_NAME
MPI_ERR_NO_MEM	MPI::ERR_NO_MEM
MPI_ERR_NOT_SAME	MPI::ERR_NOT_SAME
MPI_ERR_NO_SPACE	MPI::ERR_NO_SPACE
MPI_ERR_NO_SUCH_FILE	MPI::ERR_NO_SUCH_FILE
MPI_ERR_PORT	MPI::ERR_PORT
MPI_ERR_QUOTA	MPI::ERR_QUOTA
MPI_ERR_READ_ONLY	MPI::ERR_READ_ONLY
MPI_ERR_RMA_CONFLICT	MPI::ERR_RMA_CONFLICT
MPI_ERR_RMA_SYNC	MPI::ERR_RMA_SYNC
MPI_ERR_SERVICE	MPI::ERR_SERVICE
MPI_ERR_SIZE	MPI::ERR_SIZE
MPI_ERR_SPAWN	MPI::ERR_SPAWN
MPI_ERR_UNSUPPORTED_DATAREP	MPI::ERR_UNSUPPORTED_DATAREP
MPI_ERR_UNSUPPORTED_OPERATION	MPI::ERR_UNSUPPORTED_OPERATION
MPI_ERR_WIN	MPI::ERR_WIN
MPI_ERR_LASTCODE	MPI::ERR_LASTCODE

Buffer Address Constants

C type: `void * const`	C++ type:
Fortran type: (predefined memory location)	`void * const`
MPI_BOTTOM	MPI::BOTTOM
MPI_IN_PLACE	MPI::IN_PLACE

Assorted Constants

C type: `const int` (or unnamed `enum`)	C++ type:
Fortran type: `INTEGER`	`const int` (or unnamed `enum`)
MPI_PROC_NULL	MPI::PROC_NULL
MPI_ANY_SOURCE	MPI::ANY_SOURCE
MPI_ANY_TAG	MPI::ANY_TAG
MPI_UNDEFINED	MPI::UNDEFINED
MPI_BSEND_OVERHEAD	MPI::BSEND_OVERHEAD
MPI_KEYVAL_INVALID	MPI::KEYVAL_INVALID
MPI_LOCK_EXCLUSIVE	MPI::LOCK_EXCLUSIVE
MPI_LOCK_SHARED	MPI::LOCK_SHARED
MPI_ROOT	MPI::ROOT

Status size and reserved index values (Fortran only)

Fortran type: `INTEGER`	
MPI_STATUS_SIZE	Not defined for C++
MPI_SOURCE	Not defined for C++

MPI_TAG	Not defined for C++
MPI_ERROR	Not defined for C++

Variable Address Size (Fortran only)

Fortran type: `INTEGER`	
MPI_ADDRESS_KIND	Not defined for C++
MPI_INTEGER_KIND	Not defined for C++
MPI_OFFSET_KIND	Not defined for C++

Error-handling specifiers

C type: `MPI_Errhandler`	C++ type: `MPI::Errhandler`
Fortran type: `INTEGER`	
MPI_ERRORS_ARE_FATAL	MPI::ERRORS_ARE_FATAL
MPI_ERRORS_RETURN	MPI::ERRORS_RETURN
	MPI::ERRORS_THROW_EXCEPTIONS

Maximum Sizes for Strings

C type: `const int` (or unnamed `enum`)	C++ type:
Fortran type: `INTEGER`	`const int` (or unnamed `enum`)
MPI_MAX_PROCESSOR_NAME	MPI::MAX_PROCESSOR_NAME
MPI_MAX_ERROR_STRING	MPI::MAX_ERROR_STRING
MPI_MAX_DATAREP_STRING	MPI::MAX_DATAREP_STRING
MPI_MAX_INFO_KEY	MPI::MAX_INFO_KEY
MPI_MAX_INFO_VAL	MPI::MAX_INFO_VAL
MPI_MAX_OBJECT_NAME	MPI::MAX_OBJECT_NAME
MPI_MAX_PORT_NAME	MPI::MAX_PORT_NAME

Named Predefined Datatypes

C type: `MPI_Datatype`	C++ type: `MPI::Datatype`	**C/C++ types**
Fortran type: `INTEGER`		
MPI_CHAR	MPI::CHAR	`char` (treated as printable character)
MPI_SHORT	MPI::SHORT	`signed short int`
MPI_INT	MPI::INT	`signed int`
MPI_LONG	MPI::LONG	`signed long`
MPI_LONG_LONG_INT	MPI::LONG_LONG_INT	`signed long long`
MPI_LONG_LONG	MPI::LONG_LONG	`long long` (synonym)
MPI_SIGNED_CHAR	MPI::SIGNED_CHAR	`signed char` (treated as integral value)
MPI_UNSIGNED_CHAR	MPI::UNSIGNED_CHAR	`unsigned char` (treated as integral value)
MPI_UNSIGNED_SHORT	MPI::UNSIGNED_SHORT	`unsigned short`
MPI_UNSIGNED	MPI::UNSIGNED	`unsigned int`
		`unsigned long`

MPI_UNSIGNED_LONG	MPI::UNSIGNED_LONG	
MPI_UNSIGNED_LONG_LONG	MPI::UNSIGNED_LONG_LONG	unsigned long long
MPI_FLOAT	MPI::FLOAT	float
MPI_DOUBLE	MPI::DOUBLE	double
MPI_LONG_DOUBLE	MPI::LONG_DOUBLE	long double
MPI_WCHAR	MPI::WCHAR	wchar_t (defined in <stddef.h>) (treated as printable character)
MPI_C_BOOL	(use C datatype handle)	_Bool
MPI_INT8_T	(use C datatype handle)	int8_t
MPI_INT16_T	(use C datatype handle)	int16_t
MPI_INT32_T	(use C datatype handle)	int32_t
MPI_INT64_T	(use C datatype handle)	int64_t
MPI_UINT8_T	(use C datatype handle)	uint8_t
MPI_UINT16_T	(use C datatype handle)	uint16_t
MPI_UINT32_T	(use C datatype handle)	uint32_t
MPI_UINT64_T	(use C datatype handle)	uint64_t
MPI_AINT	(use C datatype handle)	MPI_Aint
MPI_OFFSET	(use C datatype handle)	MPI_Offset
MPI_C_COMPLEX	(use C datatype handle)	float _Complex
MPI_C_FLOAT_COMPLEX	(use C datatype handle)	float _Complex
MPI_C_DOUBLE_COMPLEX	(use C datatype handle)	double _Complex
MPI_C_LONG_DOUBLE_COMPLEX	(use C datatype handle)	long double _Complex
MPI_BYTE	MPI::BYTE	(any C/C++ type)
MPI_PACKED	MPI::PACKED	(any C/C++ type)

2.2

Named Predefined Datatypes		Fortran types
C type: MPI_Datatype	C++ type: MPI::Datatype	
Fortran type: INTEGER		
MPI_INTEGER	MPI::INTEGER	INTEGER
MPI_REAL	MPI::REAL	REAL
MPI_DOUBLE_PRECISION	MPI::DOUBLE_PRECISION	DOUBLE PRECISION
MPI_COMPLEX	MPI::F_COMPLEX	COMPLEX
MPI_LOGICAL	MPI::LOGICAL	LOGICAL
MPI_CHARACTER	MPI::CHARACTER	CHARACTER(1)
MPI_AINT	(use C datatype handle)	INTEGER (KIND=MPI_ADDRESS_KIND)
MPI_OFFSET	(use C datatype handle)	INTEGER (KIND=MPI_OFFSET_KIND)
MPI_BYTE	MPI::BYTE	(any Fortran type)
MPI_PACKED	MPI::PACKED	(any Fortran type)

C++-Only Named Predefined Datatypes C++ types

C++ type: `MPI::Datatype`	
MPI::BOOL	`bool`
MPI::COMPLEX	`Complex<float>`
MPI::DOUBLE_COMPLEX	`Complex<double>`
MPI::LONG_DOUBLE_COMPLEX	`Complex<long double>`

Optional datatypes (Fortran) Fortran types

C type: `MPI_Datatype`	C++ type: `MPI::Datatype`	Fortran type: `INTEGER`
MPI_DOUBLE_COMPLEX	MPI::F_DOUBLE_COMPLEX	`DOUBLE COMPLEX`
MPI_INTEGER1	MPI::INTEGER1	`INTEGER*1`
MPI_INTEGER2	MPI::INTEGER2	`INTEGER*8`
MPI_INTEGER4	MPI::INTEGER4	`INTEGER*4`
MPI_INTEGER8	MPI::INTEGER8	`INTEGER*8`
MPI_INTEGER16		`INTEGER*16`
MPI_REAL2	MPI::REAL2	`REAL*2`
MPI_REAL4	MPI::REAL4	`REAL*4`
MPI_REAL8	MPI::REAL8	`REAL*8`
MPI_REAL16		`REAL*16`
MPI_COMPLEX4		`COMPLEX*4`
MPI_COMPLEX8		`COMPLEX*8`
MPI_COMPLEX16		`COMPLEX*16`
MPI_COMPLEX32		`COMPLEX*32`

Datatypes for reduction functions (C and C++)

C type: `MPI_Datatype`	C++ type: `MPI::Datatype`
Fortran type: `INTEGER`	
MPI_FLOAT_INT	MPI::FLOAT_INT
MPI_DOUBLE_INT	MPI::DOUBLE_INT
MPI_LONG_INT	MPI::LONG_INT
MPI_2INT	MPI::TWOINT
MPI_SHORT_INT	MPI::SHORT_INT
MPI_LONG_DOUBLE_INT	MPI::LONG_DOUBLE_INT

Datatypes for reduction functions (Fortran)

C type: `MPI_Datatype`	C++ type: `MPI::Datatype`
Fortran type: `INTEGER`	
MPI_2REAL	MPI::TWOREAL
MPI_2DOUBLE_PRECISION	MPI::TWODOUBLE_PRECISION
MPI_2INTEGER	MPI::TWOINTEGER

Special datatypes for constructing derived datatypes

C type: `MPI_Datatype`	C++ type: `MPI::Datatype`
Fortran type: `INTEGER`	
MPI_UB	MPI::UB
MPI_LB	MPI::LB

Reserved communicators

C type: MPI_Comm	C++ type: MPI::Intracomm
Fortran type: INTEGER	
MPI_COMM_WORLD	MPI::COMM_WORLD
MPI_COMM_SELF	MPI::COMM_SELF

Results of communicator and group comparisons

C type: const int (or unnamed enum)	C++ type: const int
Fortran type: INTEGER	(or unnamed enum)
MPI_IDENT	MPI::IDENT
MPI_CONGRUENT	MPI::CONGRUENT
MPI_SIMILAR	MPI::SIMILAR
MPI_UNEQUAL	MPI::UNEQUAL

Environmental inquiry keys

C type: const int (or unnamed enum)	C++ type: const int
Fortran type: INTEGER	(or unnamed enum)
MPI_TAG_UB	MPI::TAG_UB
MPI_IO	MPI::IO
MPI_HOST	MPI::HOST
MPI_WTIME_IS_GLOBAL	MPI::WTIME_IS_GLOBAL

Collective Operations

C type: MPI_Op	C++ type: const MPI::Op
Fortran type: INTEGER	
MPI_MAX	MPI::MAX
MPI_MIN	MPI::MIN
MPI_SUM	MPI::SUM
MPI_PROD	MPI::PROD
MPI_MAXLOC	MPI::MAXLOC
MPI_MINLOC	MPI::MINLOC
MPI_BAND	MPI::BAND
MPI_BOR	MPI::BOR
MPI_BXOR	MPI::BXOR
MPI_LAND	MPI::LAND
MPI_LOR	MPI::LOR
MPI_LXOR	MPI::LXOR
MPI_REPLACE	MPI::REPLACE

Null Handles

C/Fortran name	C++ name
C type / Fortran type	C++ type
MPI_GROUP_NULL	MPI::GROUP_NULL
MPI_Group / INTEGER	const MPI::Group
MPI_COMM_NULL	MPI::COMM_NULL
MPI_Comm / INTEGER	[1])
MPI_DATATYPE_NULL	MPI::DATATYPE_NULL
MPI_Datatype / INTEGER	const MPI::Datatype
MPI_REQUEST_NULL	MPI::REQUEST_NULL
MPI_Request / INTEGER	const MPI::Request
MPI_OP_NULL	MPI::OP_NULL

MPI_Op / INTEGER	const MPI::Op
MPI_ERRHANDLER_NULL	MPI::ERRHANDLER_NULL
MPI_Errhandler / INTEGER	const MPI::Errhandler
MPI_FILE_NULL	MPI::FILE_NULL
MPI_File / INTEGER	
MPI_INFO_NULL	MPI::INFO_NULL
MPI_Info / INTEGER	const MPI::Info
MPI_WIN_NULL	MPI::WIN_NULL
MPI_Win / INTEGER	

[1]) C++ type: See Section Communicators on page Communicators regarding class hierarchy and the specific type of MPI::COMM_NULL

Empty group

C type: MPI_Group	C++ type: const MPI::Group
Fortran type: INTEGER	
MPI_GROUP_EMPTY	MPI::GROUP_EMPTY

Topologies

C type: const int (or unnamed enum)	C++ type: const int (or unnamed enum)
Fortran type: INTEGER	
MPI_GRAPH	MPI::GRAPH
MPI_CART	MPI::CART
MPI_DIST_GRAPH	MPI::DIST_GRAPH

Predefined functions

C/Fortran name	C++ name
C type / Fortran type	C++ type
MPI_COMM_NULL_COPY_FN	MPI_COMM_NULL_COPY_FN
MPI_Comm_copy_attr_function / COMM_COPY_ATTR_FN	same as in C [1])
MPI_COMM_DUP_FN	MPI_COMM_DUP_FN
MPI_Comm_copy_attr_function / COMM_COPY_ATTR_FN	same as in C [1])
MPI_COMM_NULL_DELETE_FN	MPI_COMM_NULL_DELETE_FN
MPI_Comm_delete_attr_function / COMM_DELETE_ATTR_FN	same as in C [1])
MPI_WIN_NULL_COPY_FN	MPI_WIN_NULL_COPY_FN
MPI_Win_copy_attr_function / WIN_COPY_ATTR_FN	same as in C [1])
MPI_WIN_DUP_FN	MPI_WIN_DUP_FN
MPI_Win_copy_attr_function / WIN_COPY_ATTR_FN	same as in C [1])
MPI_WIN_NULL_DELETE_FN	MPI_WIN_NULL_DELETE_FN
MPI_Win_delete_attr_function / WIN_DELETE_ATTR_FN	same as in C [1])
MPI_TYPE_NULL_COPY_FN	MPI_TYPE_NULL_COPY_FN
MPI_Type_copy_attr_function / TYPE_COPY_ATTR_FN	same as in C [1])

MPI_TYPE_DUP_FN	MPI_TYPE_DUP_FN
MPI_Type_copy_attr_function / TYPE_COPY_ATTR_FN	same as in C [1])
MPI_TYPE_NULL_DELETE_FN	MPI_TYPE_NULL_DELETE_FN
MPI_Type_delete_attr_function / TYPE_DELETE_ATTR_FN	same as in C [1])

[1] See the advice to implementors on MPI_COMM_NULL_COPY_FN, ... in Section Communicators on page Communicators

Deprecated predefined functions

C/Fortran name	C++ name
C type / Fortran type	C++ type
MPI_NULL_COPY_FN	MPI::NULL_COPY_FN
MPI_Copy_function / COPY_FUNCTION	MPI::Copy_function
MPI_DUP_FN	MPI::DUP_FN
MPI_Copy_function / COPY_FUNCTION	MPI::Copy_function
MPI_NULL_DELETE_FN	MPI::NULL_DELETE_FN
MPI_Delete_function / DELETE_FUNCTION	MPI::Delete_function

Predefined Attribute Keys

C type: const int (or unnamed enum)	C++ type:
Fortran type: INTEGER	const int (or unnamed enum)
MPI_APPNUM	MPI::APPNUM
MPI_LASTUSEDCODE	MPI::LASTUSEDCODE
MPI_UNIVERSE_SIZE	MPI::UNIVERSE_SIZE
MPI_WIN_BASE	MPI::WIN_BASE
MPI_WIN_DISP_UNIT	MPI::WIN_DISP_UNIT
MPI_WIN_SIZE	MPI::WIN_SIZE

Mode Constants

C type: const int (or unnamed enum)	C++ type:
Fortran type: INTEGER	const int (or unnamed enum)
MPI_MODE_APPEND	MPI::MODE_APPEND
MPI_MODE_CREATE	MPI::MODE_CREATE
MPI_MODE_DELETE_ON_CLOSE	MPI::MODE_DELETE_ON_CLOSE
MPI_MODE_EXCL	MPI::MODE_EXCL
MPI_MODE_NOCHECK	MPI::MODE_NOCHECK
MPI_MODE_NOPRECEDE	MPI::MODE_NOPRECEDE
MPI_MODE_NOPUT	MPI::MODE_NOPUT
MPI_MODE_NOSTORE	MPI::MODE_NOSTORE
MPI_MODE_NOSUCCEED	MPI::MODE_NOSUCCEED
MPI_MODE_RDONLY	MPI::MODE_RDONLY
MPI_MODE_RDWR	MPI::MODE_RDWR
MPI_MODE_SEQUENTIAL	MPI::MODE_SEQUENTIAL
MPI_MODE_UNIQUE_OPEN	MPI::MODE_UNIQUE_OPEN
MPI_MODE_WRONLY	MPI::MODE_WRONLY

Datatype Decoding Constants

C type: `const int` (or unnamed `enum`) Fortran type: `INTEGER`	C++ type: `const int` (or unnamed `enum`)
MPI_COMBINER_CONTIGUOUS	MPI::COMBINER_CONTIGUOUS
MPI_COMBINER_DARRAY	MPI::COMBINER_DARRAY
MPI_COMBINER_DUP	MPI::COMBINER_DUP
MPI_COMBINER_F90_COMPLEX	MPI::COMBINER_F90_COMPLEX
MPI_COMBINER_F90_INTEGER	MPI::COMBINER_F90_INTEGER
MPI_COMBINER_F90_REAL	MPI::COMBINER_F90_REAL
MPI_COMBINER_HINDEXED_INTEGER	MPI::COMBINER_HINDEXED_INTEGER
MPI_COMBINER_HINDEXED	MPI::COMBINER_HINDEXED
MPI_COMBINER_HVECTOR_INTEGER	MPI::COMBINER_HVECTOR_INTEGER
MPI_COMBINER_HVECTOR	MPI::COMBINER_HVECTOR
MPI_COMBINER_INDEXED_BLOCK	MPI::COMBINER_INDEXED_BLOCK
MPI_COMBINER_INDEXED	MPI::COMBINER_INDEXED
MPI_COMBINER_NAMED	MPI::COMBINER_NAMED
MPI_COMBINER_RESIZED	MPI::COMBINER_RESIZED
MPI_COMBINER_STRUCT_INTEGER	MPI::COMBINER_STRUCT_INTEGER
MPI_COMBINER_STRUCT	MPI::COMBINER_STRUCT
MPI_COMBINER_SUBARRAY	MPI::COMBINER_SUBARRAY
MPI_COMBINER_VECTOR	MPI::COMBINER_VECTOR

Threads Constants

C type: `const int` (or unnamed `enum`) Fortran type: `INTEGER`	C++ type: `const int` (or unnamed `enum`)
MPI_THREAD_FUNNELED	MPI::THREAD_FUNNELED
MPI_THREAD_MULTIPLE	MPI::THREAD_MULTIPLE
MPI_THREAD_SERIALIZED	MPI::THREAD_SERIALIZED
MPI_THREAD_SINGLE	MPI::THREAD_SINGLE

File Operation Constants, Part 1

C type: `const MPI_Offset` (or unnamed `enum`) Fortran type: `INTEGER (KIND=MPI_OFFSET_KIND)`	C++ type: `const MPI::Offset` (or unnamed `enum`)
MPI_DISPLACEMENT_CURRENT	MPI::DISPLACEMENT_CURRENT

File Operation Constants, Part 2

C type: `const int` (or unnamed `enum`) Fortran type: `INTEGER`	C++ type: `const int` (or unnamed `enum`)
MPI_DISTRIBUTE_BLOCK	MPI::DISTRIBUTE_BLOCK
MPI_DISTRIBUTE_CYCLIC	MPI::DISTRIBUTE_CYCLIC
MPI_DISTRIBUTE_DFLT_DARG	MPI::DISTRIBUTE_DFLT_DARG
MPI_DISTRIBUTE_NONE	MPI::DISTRIBUTE_NONE
MPI_ORDER_C	MPI::ORDER_C
MPI_ORDER_FORTRAN	MPI::ORDER_FORTRAN
MPI_SEEK_CUR	MPI::SEEK_CUR
MPI_SEEK_END	MPI::SEEK_END
MPI_SEEK_SET	MPI::SEEK_SET

F90 Datatype Matching Constants

C type: `const int` (or unnamed `enum`)	C++ type:
Fortran type: `INTEGER`	`const int` (or unnamed `enum`)
MPI_TYPECLASS_COMPLEX	MPI::TYPECLASS_COMPLEX
MPI_TYPECLASS_INTEGER	MPI::TYPECLASS_INTEGER
MPI_TYPECLASS_REAL	MPI::TYPECLASS_REAL

2.2

Constants Specifying Empty or Ignored Input

C/Fortran name	C++ name
C type / Fortran type	C++ type
MPI_ARGVS_NULL	MPI::ARGVS_NULL
`char***` / 2-dim. array of `CHARACTER*(*)`	`const char ***`
MPI_ARGV_NULL	MPI::ARGV_NULL
`char**` / array of `CHARACTER*(*)`	`const char **`
MPI_ERRCODES_IGNORE	Not defined for C++
`int*` / `INTEGER` array	
MPI_STATUSES_IGNORE	Not defined for C++
`MPI_Status*` / `INTEGER, DIMENSION(MPI_STATUS_SIZE,*)`	
MPI_STATUS_IGNORE	Not defined for C++
`MPI_Status*` / `INTEGER, DIMENSION(MPI_STATUS_SIZE)`	
MPI_UNWEIGHTED	Not defined for C++

C Constants Specifying Ignored Input (no C++ or Fortran)

C type: `MPI_Fint*`
MPI_F_STATUSES_IGNORE
MPI_F_STATUS_IGNORE

C and C++ preprocessor Constants and Fortran Parameters

C/C++ type: `const int` (or unnamed `enum`)
Fortran type: `INTEGER`
MPI_SUBVERSION
MPI_VERSION


Up: Defined Values and Handles **Next:** Types **Previous:** Defined Values and Handles

Return to MPI-2.2 Standard Index
Return to MPI Forum Home Page

(Unofficial) MPI-2.2 of September 4, 2009
HTML Generated on September 10, 2009