

1	MPI_BAND	bit-wise and
2	MPI_LOR	logical or
3	MPI_BOR	bit-wise or
4	MPI_LXOR	logical exclusive or (xor)
5	MPI_BXOR	bit-wise exclusive or (xor)
6	MPI_MAXLOC	max value and location
7	MPI_MINLOC	min value and location

8 The two operations MPI\_MINLOC and MPI\_MAXLOC are discussed separately in Sec-  
 9 tion 5.9.4. For the other predefined operations, we enumerate below the allowed combi-  
 10 nations of `op` and `datatype` arguments. First, define groups of MPI basic datatypes in the  
 11 following way.

13		
14	C integer:	MPI_INT, MPI_LONG, MPI_SHORT,
15		MPI_UNSIGNED_SHORT, MPI_UNSIGNED,
16		MPI_UNSIGNED_LONG,
17		MPI_LONG_LONG_INT,
18		MPI_LONG_LONG (as synonym),
19		MPI_UNSIGNED_LONG_LONG,
20		MPI_SIGNED_CHAR, MPI_UNSIGNED_CHAR
ticket64. 21	Fortran integer:	MPI_INTEGER and handles returned from
ticket64. 22		MPI_TYPE_CREATE_F90_INTEGER,
23		and if available: MPI_INTEGER1,
24		MPI_INTEGER2, MPI_INTEGER4,
25		MPI_INTEGER8, MPI_INTEGER16
26	Floating point:	MPI_FLOAT, MPI_DOUBLE, MPI_REAL,
27		MPI_DOUBLE_PRECISION
ticket64. 28		MPI_LONG_DOUBLE
29		and handles returned from
30		MPI_TYPE_CREATE_F90_REAL,
31		and if available: MPI_REAL2,
32		MPI_REAL4, MPI_REAL8, MPI_REAL16
33	Logical:	MPI_LOGICAL
ticket64. 34	Complex:	MPI_COMPLEX and handles returned from
ticket64. 35		MPI_TYPE_CREATE_F90_COMPLEX
36		and if available: MPI_DOUBLE_COMPLEX,
37		MPI_COMPLEX4, MPI_COMPLEX8,
38		MPI_COMPLEX16, MPI_COMPLEX32
39	Byte:	MPI_BYTE

40 Now, the valid datatypes for each option is specified below.

41		
42	Op	Allowed Types
43		
44	MPI_MAX, MPI_MIN	C integer, Fortran integer, Floating point
45	MPI_SUM, MPI_PROD	C integer, Fortran integer, Floating point, Complex
46	MPI_LAND, MPI_LOR, MPI_LXOR	C integer, Logical
47	MPI_BAND, MPI_BOR, MPI_BXOR	C integer, Fortran integer, Byte
48		