MPI_DIST_GRAPH_CREATE, the constants MPI_UNWEIGHTED, and the derived C++ class Distgraphcomm were added. ticket33.

18. Section 7.5.4 on page 262.
For the scalable distributed graph topology interface, the functions MPI_DIST_NEIGHBORS_COUNT and MPI_DIST_NEIGHBORS and the constant MPI_DIST_GRAPH were added. ticket3.

19. Section 7.5.4 on page 262.
Remove ambiguity regarding duplicated neighbors with MPI_GRAPH_NEIGHBORS and MPI_GRAPH_NEIGHBORS_COUNT. ticket101.

20. Section 8.1.1 on page 273.
The subversion number changed from 1 to 2. ticket7.

21. Section 8.3 on page 278, Section ?? on page ??, and Annex A.1.3 on page 519.
Changed function pointer typedef names MPI_{Comm,File,Win}_errhandler_fn to MPI_{Comm,File,Win}_errhandler_function. Deprecated old "_fn" names. ticket71.

22. Section 8.7.1 on page 297.
Attribute deletion callbacks on MPI_COMM_SELF are now called in LIFO order. Implementors must now also register all implementation-internal attribute deletion callbacks on MPI_COMM_SELF before returning from MPI_INIT/MPI_INIT_THREAD. ticket43.

23. Section 11.3.4 on page 347.
The restriction added in MPI 2.1 that the operation MPI_REPLACE in MPI_ACCUMULATE can be used only with predefined datatypes has been removed. MPI_REPLACE can now be used even with derived datatypes, as it was in MPI 2.0. Also, a clarification has been made that MPI_REPLACE can be used only in MPI_ACCUMULATE, not in collective operations that do reductions, such as MPI_REDUCE and others. ticket6.

24. Section 12.2 on page 373.
Add "*" to the query_fn, free_fn, and cancel_fn arguments to the C++ binding for MPI::Grequest::Start() for consistency with the rest of MPI functions that take function pointer arguments. ticket18.

25. Section 13.5.2 on page 430, and Table 13.2 on page 432.
MPI_(U)INT{8,16,32,64}_T, MPI_AINT, MPI_OFFSET, MPI_C_COMPLEX, MPI_C_FLOAT_COMPLEX, MPI_C_DOUBLE_COMPLEX, MPI_C_LONG_DOUBLE_COMPLEX, and MPI_C_BOOL are added as predefined datatypes in the external32 representation. ticket55.

26. Section 16.3.7 on page 502.
**!!!TODO!!! See Ticket – proposed text:** The description was modified that it only describes how an MPI implementation behaves, but not how it must be implemented internally. The erroneous MPI-2.1 Example 16.17 was replaced with three new examples **[...insert reference to example numbers...]** on page **[...pageref...]** explicitly detailing cross-language attribute behavior. Implementations that matched the behavior of the old example will need to be updated. ticket4.