Chapter 2: MPI Terms and Conventions
MPI-4.0 → MPI-4.1-RC

Chapter Committee:

- Claudia Blaas-Schenner
- Bill Gropp
- Tony Skjellum
- Puri Bangalore
- Guillaume Mercier
- Dan Holmes
- Julien Jaeger

MPI Forum Meeting (RCM for MPI 4.1), September 2023, Bristol, UK

Chapter 2: MPI Terms and Conventions changes: MPI-4.0 → MPI-4.1-RC

- # 23 (PR #708): Allow MPI_WIN_SHARED_QUERY on created and allocated windows (2nd on 2023-05)
- # 57 (PR #790): Deprecate MPI_HOST (2nd on 2023-05)
- #471 (PR # 751): Fix Incorrect Usage of Rank/Task/etc. in Terms Chapter (2nd on 2023-03)
- #493 (PR # 753): Issue N - Progress-Section in terms-2.tex - for RC-4-June (2nd on 2023-03)
- #518 (PR # 623 or 829): Deprecate all _x procedures (2nd on 2023-05)
- #561 (PR # 722): Deprecate mpif.h (2nd on 2023-03)
- #564 (PR #800): Fortran Binding Issues and Capitalization (**errata** on 2023-03)
- #565 (PR # 727): Refer to Fortran 2018 instead of 2008+TS (2nd on 2023-03)
- #581 (PR #665): ULFM Fault Tolerance (slice 1: ack_failed, get_failed, revoke) (2nd on 2023-02) → **MPI 5.0**
- #586 (PR #736): MPI_Buffer_attach and Sessions (2nd on 2023-07)
- #588 (PR # 644): Clarification of errhandler fallback with Sessions (2nd on 2023-05)
- #641 (PR # 752): Including Annex A.2 back into the standard (2nd on 2023-03)
- #646 (PR # 771): WTIME and WTICK should be symbols (2nd on 2023-03)
- #647 (PR # 770): Handle conversions should be symbols (2nd on 2023-03)
- #658 (PR #776): MPI_MAX_STRINGTAG_LEN and MPI_MAX_PSET_NAME_LEN arent listed as compile-time constants (**errata** on 2023-05) → #705 (PR #822)
- #669 (PR # 788): Add operation state 'enabled' and 'local calls' into Terms (2nd on 2023-07)
- #676 (PR #825): 'Pending operation' not defined, pending proper definition (**errata** on 2023-07)
- #679 (PR #820): Errata: Noncollective (for procedure) and nonpersistent are not defined (**errata** on 2023-07)
- #705 (PR #822): Errata: Fortran has only compile-time constants (**errata** on 2023-07**)**
- PR #863: Pre-RC4.1 fixes in the Terms chapter (**chapter committee**)

20 merged
14 2nd votes
5 errata
1 cc (PR only)

Chapter 2: MPI Terms and Conventions reviewer comments

- **Chapter 2 =** PDF 51-72 / **MPI-4.1 9-30**
- **Aurelien Bouteiller -** PDF 51-60 / **MPI-4.1 9-18**
- **Benson Muite -** PDF 60- 72 / **MPI-4.1 18-30**
- `MPI bindings are for Fortran 90 or later...` should this be `MPI bindings are for Fortran 90 and for Fortran 2008 or later...` p 22 line 33
- May want to have Table 2.1 one page earlier on pg 23 where it is first mentioned rather than on p 24
- May want to indicate an example mpi.h header file is provided with the standard. p 25 line 3
- `an MIMD style` should probably be `a MIMD style` p 25 line 24
  (The acronym MIMD - Multiple Instance, Multiple Data is not expanded the first time MIMD is used on p 5 line 37)

Chapter 2: MPI Terms and Conventions

TODO

- **page numbers as they are printed in the document**
- **Fortran**
- page 11 / line 16:

  “Fortran” in this document refers to Fortran 90 and higher; see Section 2.6

  → write “Fortran 90 or later” to be consistent with page 22
- page 22 / line 33:

  (`MPI bindings are for Fortran 90 or later...` should this be `MPI bindings are for Fortran 90 and for Fortran 2008 or later...`)

  MPI bindings are for Fortran 90 or later, though they were originally designed to be usable in Fortran 77 environments.

  → don’t change
- page 23 / lines 21-22:

  2.6.2 Fortran Binding Issues

  Originally, MPI-1.1 provided bindings for Fortran 77. These bindings are retained, but they are now interpreted in the context of the Fortran 90 standard. MPI can still be used with most Fortran 77 compilers, as noted below. When the term “Fortran” is used it means Fortran 90 or later; it means Fortran 2008 with TS 29113, which is now an integral part of Fortran 2018 and later if the mpi_f08 module is used.

  → add a comma “later, ...”

  or even better rewrite the last part of the sentence: “if the mpi_f08 module is used it means Fortran 2008 ...”

Chapter 2: MPI Terms and Conventions TODO

- **page numbers as they are printed in the document**
- page 12 / line 41:
  Additionally, an MPI operation can be collective or noncollective.
  → Can we have some vertical space above this sentence (and maybe a \noindent), it starts discussing a new idea (”subsection”).
- page 12 / line 43-44:
  Collective operation: A set of related operations, one per MPI process in a group or groups of MPI processes.
  → Collective operations are a set...
- page 13 / line 10-11:
  Noncollective operation: Noncollective operations are...
  → write in a consistent style with above
- page 13 / line 20:
  Enabled An MPI operation...
  → “Enabled:”
- page 14 / line 35:
  The following are properties of MPI operation-related procedures:
  → Can we have some vertical space above this sentence (and maybe a \noindent), it starts discussing a new idea (”subsection”).

Chapter 2: MPI Terms and Conventions

TODO

- **page numbers as they are printed in the document**

- page 12:

  → Figure 2.3 should be on page 12 (instead of page 13) to have the 3 figures and the description in the text on the same page

- page 23:

  (May want to have Table 2.1 one page earlier on pg 23 where it is first mentioned rather than on p 24)

  → Table 2.1 should be on page 23 (instead of page 24)

- page 25 / line 3:

  2.6.3 C Binding Issue ...

  The definition of named constants, function prototypes, and type definitions must be supplied in an include file mpi.h.

  (May want to indicate an example mpi.h header file is provided with the standard. p 25 line 3)

  → ?

- page 25 / line 24:

  `an MIMD style` should probably be `a MIMD style` p 25 line 24

  (The acronym MIMD - Multiple Instance, Multiple Data is not expanded the first time MIMD is used **on p 5 line 37**)

  → ?, expand the acronym