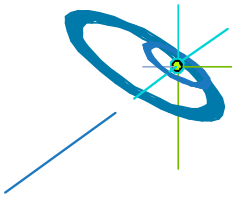
MPI 2.1
at
MPI Forum
Chicago, March 10-12, 2008

Rolf Rabenseifner
rabenseifner@hlrs.de
(Chairman of MPI 2.1 Task)

University of Stuttgart
High-Performance Computing-Center Stuttgart (HLRS)
www.hlrs.de



Working plan: March 10-12, 2008

Monday:	2:00 pm - 3:00 pm : MPI 2.1 -	Ballot 4 proposals	B4
	3:15 pm - 5:30 pm : MPI 2.1 -	Ballot 4 proposals – continued	
	5:15 pm - 5:30 pm : MPI 2.1 -	Ballot 3 preparation of first vote	B3
	9:00 pm - 10:00 pm: MPI 2.1 -	**Ballot 4, additional discussions**	
		(End of Ballot 4 discussions)	
Tuesday:	9:00 am - 9:30 am : MPI 2.1 -	Ballot 4: Straw vote	B4
	9:30 am - 10:00 am : MPI 2.1 -	Ballot 3: First official vote	B3
		(Was discussed at Jan. 2008 meeting)	
	1:30 pm - 3:00 pm : MPI 2.1 -	Combined document review	2.1
	3:15 pm - 3:45 pm : MPI 1.3 -	Review	1.3
Wednesday:	11:00 am - 11:30 am : MPI 2.1 -	Follow up	>

MPI-1.3 – Working Plan

- MPI-1.3 draft Mar 04, 2008 is available
- Reviews are done / will be done by the MPI 1.3 reviewing group:
  1. **Bill Gropp** (@meeting Jan-2008)
  2. **Rolf Rabenseifner** (@meeting Jan-2008)
  3. **Adam Moody** (@meeting Jan-2008)
  4. **Puri Bangalore** (@meeting Jan-2008)
  5. **Terry Dontje** (@meeting Mar-2008)
  6. **William Yu** (not @meeting Jan-2008)
- In the final version of MPI-1.3, also the MPI-2.1 Ballot 4 items 5, 10.e, 14, and 15 will be included (if voted positive, March 11)
- Based on current available reviews, final version will be done until Mar 16, 2008
- Discussion only if differences between views of reviewers & editor
- Final review should be done until Mar 23, 2008. Okay **?**
- If there are still some open issues → reiteration
- Final version → Official reading at April, 1st vote June, 2nd vote Sep.

MPI-2.1 – Working Plan

Monday:

- All reviewer are asked
  - to check whether all editor's decisions about their review are okay:

    OK = Obvious, typo, correction must be done, ...
    or good idea, correction should be done, ...

    ? = should be decided by the MPI Forum

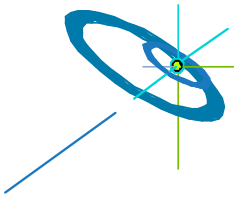
    22 = should be deferred to MPI-2.2.
    It is mainly a suggestion to the content of MPI and not to the merging of MPI-1.1 with MPI-2.

    Q? = Question & answer only, no need for corrections

    - = Comment, can be ignored

    ***NO* = Correction must or should not be done**
  - OK-items = same opinion of editor and reviewer (~100 items)
    → expectation: no discussion need → should be okay for the forum
  - Especially the *NO* and "?" must be checked and discussed, probably off-line

Tuesday: Remaining "?" must be decided by the Forum, see **review.txt**



Follow-Up: Next Steps

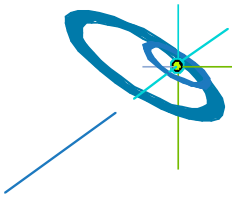
- Ballot 3 → Apr'08 2nd vote = done
- Ballot 4 → Apr'08 1st vote → Jun'08 2nd vote = done
- MPI-1.3 → 2nd draft Mar.16 → **group of reviewer**
  - **Puri Bangalore**
  - **Terry Dontje**
  - **Bill Gropp**
  - **Adam Moody**
  - **Rolf Rabenseifner**

  → **review until Mar.23**

  → re-iteration (only if necessary)

  → official reading at Apr.2008 meeting

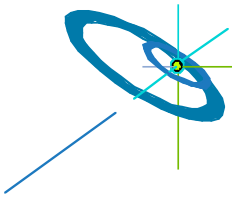
  → Jun'08 1st vote → Sep'08 2nd vote = done



Next Steps – MPI-2.1

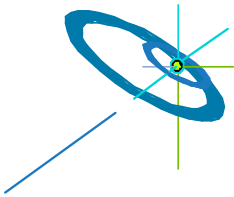
- Input:
  - **Final MPI-1.3 (from Rainer Keller, Mar. 16, 2008)**
  - **MPI-1.2 C++ interfaces on single lines (from Jeff Squyres, Mar. 20, 2008 ???)**
  - **Merged Language binding Annexes to one Annex A with**
    - Section A.1 Fortran Binding
    - Section A.2 C Binding
    - Section A.3 C++ Binding

    (from Alexander Supalov, Mar. ???, 2008)
  - **List of Examples (from Rainer Keller, Mar. 20, 2008)**
  - **Acknowledgements (from Richard Graham, Mar. 20, 2008)**
  - **Change Log (from Rolf Rabenseifner, Mar. 18, 2008)**
  - **Check of "removed text" (from Bill Gropp, Mar 12, 2008)**
- First output:
  - Draft (Mar. 29, 2008)



Next Steps – MPI-2.1 (continued)

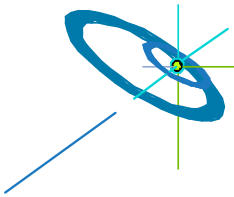
- MPI-2.1
  - → handed over **to the MPI-2.1 chapter authors (Mar. 29, 2008)**
  - → **Implementing merged document goals (14 days, until April 13, 2008)**
  - → Review (7 days, until April 20)
  - → Re-iteration of details detected by reviewers (5 days, until April 25)
  - → April 28-30, 2008: Meeting in Chicago
  - → Official reading (review report)
  - → Jun'08 1st vote
  - → Sep'08 2nd vote = done



We need reviewers for: (**bold=large**) : **(red=responsible chapter author)** (**Reviewer, green=@meeting**)

Frontmatter	**Bill Gropp, Rusty Lusk**
Chap. 1: Introduction to MPI	**Bill Gropp, Rusty Lusk, Karl Feind, Adam Moody, Traeff**
Chap. 2: MPI-2 Terms and Conventions	**Tony Skjellum, Bill Gropp, Richard Barrett, Traeff**
Chap. 3: Point-to-Point Communication (incl. sections from MPI-2 Misc. + 8.9)	**Rich Graham, Jespar Larsson Traeff, George Bosilca, Steve Poole, Kannan Narasimhan, David Solt, B. Gropp Matt Koop, Adam Moody**
Chap. 4: Collective Communication (incl. sections from MPI-2 Ext. Collect.)	**Adam Moody,** Steven Ericsson-Zenith, **Edgar Gabriel, R. Thakur, B. Gropp, G. Bosilca, Th. Hoefler, J. Traeff**
Chap. 5: Groups, Context, and Communicators (incl. sections from MPI-2 Ext.Col. + 8.8)	**Richard Treumann** Steven Ericsson-Zenith, **Edgar Gabriel, Tony Skjellum, Bill Gropp, G. Bosilca, Robert Blackmore**
Chap. 6: Process Topologies	**Jesper L. Traeff, Rusty Lusk, Bill Gropp, Richard Barrett**
Chap. 7: MPI Environmental Management (incl. sections from MPI-2 Misc.)	**George Bosilca, Rich Graham, Jespar Larsson Traeff, Steve Poole, Kannan Narasimhan, David Solt, B. Gropp**
Chap. 8: Miscellany	**Jesper L. Traeff, Rich Graham, George Bosilca, Steve Poole, Kannan Narasimhan, B. Gropp**
Chap. 9: Process Creation and Management	**David Solt,** Dries Kimpe, **Rusty Lusk, George Bosilca, Bill Gropp, Kalem Karian,**
Chap. 10: One-Sided Communication	**Jespar Larsson Traeff**, Ericsson-Zenith, Martin Schulz, **Bill Gropp, Darius Buntinas,**
Chap. 11: External Interfaces	**Bronis de Supinski, Bill Gropp, Rainer Keller**
Chap. 12: I/O	**Rajeev Thakur**, Joachim Worringen, **Bill Gropp, Koziol**
Chap. 13: Language Bindings	**Jeff Squyres, Steve Poole, Purushotham Bangalore, Bill Gropp, Erez Haba, Alexander Supalov**
Chap. 14: Profiling Interface	**Bronis de Supinski, Bill Gropp, Jeff Brown**
Chap. 15: Deprecated Functions	**Rolf Rabenseifner**
Bibliography	**Bill Gropp, Rusty Lusk**
Annex A Language Bindings	**A. Supalov, J. Squyres, St. Poole, P. Bangalore, B. Gropp**

Appendix: Slides from January 2008 Meeting



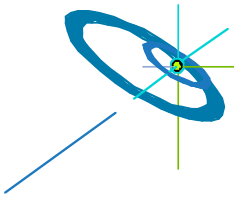
MPI 2.1 Goals

Scope of Effort:

- Clarification to the MPI standards document,
- resulting in a single document describing the full MPI 2.1 standard.
- This includes merging of documents, text corrections, and added clarifying text.

Working plan:

- MPI 1.1 **+** Chap. 3 of MPI-2 (Version 1.2 of MPI) + some errata will be combined to
  → **MPI 1.3**
- MPI 1.2.1 **+** rest of MPI-2 (MPI 2.0) will be combined to
  → **MPI 2.1 draft (without clarifications)**
- adopted MPI 2.1 Ballots 1&2 **+** new MPI 2.1 ballots 3&4 are combined to the
  → **Ballot 1-4 of MPI 2.1 adopted errata**
  (with references still based on MPI 1.1 and MPI-2 documents)
- MPI 2.1 draft **+** MPI 2.1 adopted errata
  → **MPI 2.1**

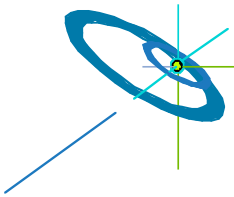


Why combined documents?

The goals behind this combining of the documents have been already expressed in the MPI-1.1 standard:

"Sect. 1.2 Who should use this standard?
This standard is intended for use by all those who want to write portable message-passing programs in Fortran 77 and C.
This includes individual application programmers, developers of software designed to run on parallel machines, and creators of environments and tools. ..."

It is more efficient that the MPI Forum combines the documents once than every user of the MPI documents has to do this in his/her daily work based on the combination of MPI-1.1 and the several updating documents, i.e., MPI-2, and the future updates 2.1, 2.2, ...



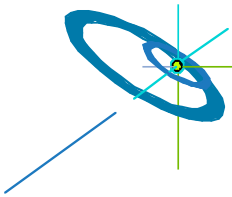
Schedule based on official rules

Rules and Procedures

1. Here is a reminder of the traditional MPI voting rules, which have served us well. These rules have been extended to the email discussion of MPI erratas and have been applied to the errata ballots. We expect to adapt these rules, preserving their spirit, as we go forward.
2. One vote per organization
3. To vote, an organization must have been present at the last two MPI Forum meetings.
4. **Votes are taken twice, at separate meetings. Votes are preceded by a reading at an earlier meeting, to familiarize everyone with the issues.**
5. Measures pass on a simple majority.
6. Only items consistent with the charter can be considered.

From **http://www.mpi-forum.org/mpi2_1/index.htm**

For **MPI x.x combined documents**:
This reading at the MPI Forum meetings will be substituted by a review report through a review group. Each Forum member can be part of this group.
With the 1st official vote on a combined document (at next meeting), this modification of the voting rules is accept for that document.



MPI 2.1 Review Procedure (Voting plan) – Step 1

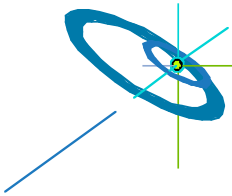
action point

- Straw **vote** on the working plan (see 4 steps on previous slide)

- MPI 1.1 **+** Chap. 3 of MPI-2 (Version 1.2 of MPI) + some errata will be combined to → **MPI 1.3**

action point

  - Jan.08 meeting:
    Short discussion and **defining a review group** who is reviewing the **MPI 1.3 merging plan** (printed copies available) and the **MPI 1.3 combined document**
  - See e-mail: From: Rainer Keller, Subject: Re: [mpi-21] Documents
    Date: Mon, 7 Jan 2008 12:13:14 +0100
  - Reporting by e-mail on mpi-21 reflector
  - Corrections if necessary (until Jan. 31, 2008)
    → final version of **MPI 1.3 merging plan** and **MPI 1.3**
  - Final report of the reviewers at March 2008 meeting (=substitutes the **reading**)
  - **1st vote** by the MPI Forum at April 2008 meeting
  - **2nd (final) vote** by the MPI Forum at June 2008 meeting



MPI 2.1 Review Procedure (Voting plan) – Step 2

- MPI 1.3 combined document **+** rest of MPI-2 (MPI 2.0) will be combined to
  → **MPI 2.1 draft**
  - **Discussion of the 11 major merging decisions** and finishing them with **straw votes** (**Jan.2008** meeting)
    based on the distributed text (printed copies available)

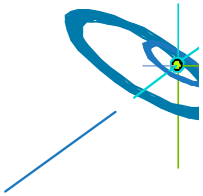
  - **Defining a review group** (**Jan.2008** meeting)

  - First draft of combined document (Feb 22, 2008, to be done by Rolf Rabenseifner)
  - Reviewing process and report of the reviewers (until March 10-12, 2008 meeting)
  - Discussion and further corrections if necessary (March 2008 meeting)
  - All necessary straw votes should be done at end of March 2008 meeting.
  - April 1, 2008, the final document should be available for twice voting.
  - Final report of the reviewers at April 2008 meeting (=substitutes the **reading**)
  - **1st vote** by the MPI Forum at June 2008 meeting
  - **2nd (final) vote** by the MPI Forum at Sep. 2008 meeting

MPI 2.1 Review Procedure (Voting plan) – Step 3

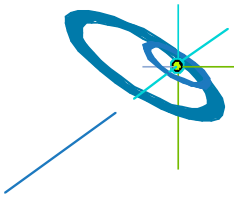
- adopted MPI 2.1 Ballots 1&2 **+** new MPI 2.1 ballots 3&4 are combined to the
  → **MPI 2.1 adopted errata**
  (with references still based on MPI 1.1 and MPI-2 documents)
  - Ballots 1&2 are done (Chapter 1, Errata for MPI-2, May 15, 2002)
    http://www.mpi-forum.org/docs/errata-20-2.pdf
  - Ballot 3
    http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/index.html
    http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/ballot3.html

  - **Official Reading** and straw votes (Jan. 14, 2008)
    Re-reading if modifications are necessary (Jan. 15 or 16, 2008)
  - **First official vote** (on all topics with positive straw votes) (March, 2008)
  - Ballot 4 at March 2008 meeting with official reading (1st day and next days).
    Unresolved items are moved to MPI 2.2.
  - → **Final text of Ballot 3+4 is available at end of March meeting,**
    i.e., it can be already included into MPI 2.1 combined document.
  - April 2008: Ballot 3: 2nd vote; Ballot 4: 1st vote
  - June 2008: Ballot 4: 2nd vote (all second votes are based on text in the combined documents) → **MPI 2.1 adopted errata** finished



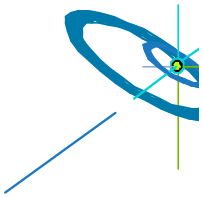
MPI 2.1 Review Procedure (Voting plan) – Step 4

- MPI 2.1 draft **+** MPI 2.1 adopted errata
  → **MPI 2.1 combined document**
  - The MPI 2.1 Ballots 1-4 (as after final reading on April 2008 meeting) are included into the MPI 2.1 draft (from April 1, 2008) (as prepared for final review/reading at April 2008 meeting)
    → MPI 2.1 combined document (April 14, 2008)
  - Defining the reviewing group (on March 2008 meeting) (may be smaller as for the MPI2.1 draft)
  - Reporting by e-mail on mpi-21 reflector until April 18, 2008
  - Corrections if necessary until April 23, 2008
  - Final report of the reviewers at April 2008 meeting (=substitutes the **reading**)
  - **First vote** on June 2008 meeting
  - **Second (final) vote** on Sep. 2008 meeting



Schedule on this meeting

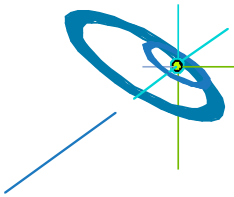
- Monday, Jan 14, 2008
  - 2:15pm - 3:15pm : MPI 2.1 Introduction
    - **Action point: Discussion and straw vote on working plan**
    - **Starting with Reading of Ballot 3**
  - 3:30pm - 5:00pm : Ballot 3 + Combined documents
    - **Reading of Ballot 3**
    - **MPI 1.3 combined document issues → defining the review group (≥ 3 persons)**
    - **MPI 2.0 combined document issues, discussions and straw votes**
    - **Defining the MPI 2.0 combined document review group (≥ 2 per chapter)**
- Tuesday, Jan 15, 2008
  - 9:00am - 9:30am: Discussion and straw vote on Ballot 3
    - **"Last" chance for veto on details of Ballot 3 ☺**
    - **Reading of modified items + straw votes**
- Wednesday, Jan 16, 2008
  - 9:00am - 10:30am : First vote of MPI 2.1 Ballot 3
    - **"Very last" chance for veto on details of Ballot 3 ☺**
    - **Final discussions and decisions on MPI 1.3 and MPI 2.0 combined documents**
  - 10:30am - 10:40am : Report Back from Committees: MPI 2.1 (Rolf)



Tough schedule

- I expect that most or all Ballot 3 issues are obvious.
- Therefore I hope that we can spent most time on the technical merging aspects
  - MPI 1.3 merging
  - MPI 2.0 merging

- Overall schedule 6 min / slide
  - Longer discussions may be deferred
    - **MPI 2.1 problems → MPI 2.2**
    - **Merging problems → evening discussion in a subgroup of the Forum**
  - My goal: to fix all, that can be fixed already, because it is trivial, obvious, consensus, … …



Discussion on working plan

Question:

Do we want to produce the combined docu-ments?

Yes:

All (=43)

No:

0

Abstain:

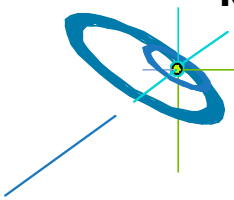
0

Working plan:

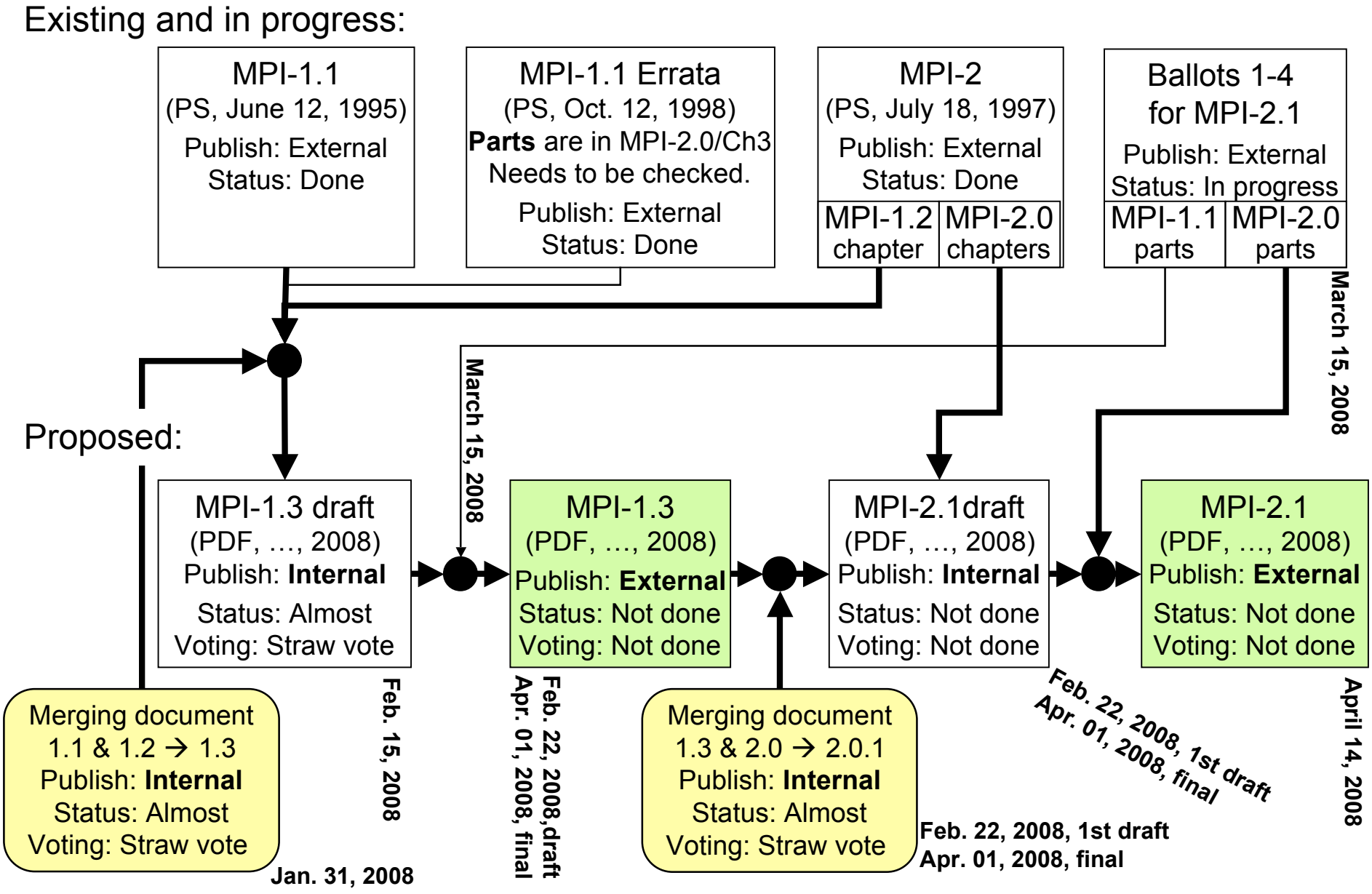
- MPI 1.1 **+** Chap. 3 of MPI-2 (Version 1.2 of MPI) + some errata will be combined to
  → **MPI 1.3 combined document** **(**work is already mainly done by Rainer Keller**)**
- MPI 1.3 combined document **+** rest of MPI-2 (MPI 2.0) will be combined to
  → **MPI 2.1 draft (combined doc.)** **(**work is already 50% done by Rolf Rabenseifner**)**
- adopted MPI 2.1 Ballots 1&2 **+** new MPI 2.1 ballots are combined to the
  → **Ballot 1-4 for MPI 2.1 adopted errata**
  (with references still based on MPI 1.1 and MPI-2 documents)
- MPI 2.1 draft **+** MPI 2.1 adopted errata
  → **MPI 2.1 (final combined document)**
  (small extra work, because errata show exact locations**)**

Comments?

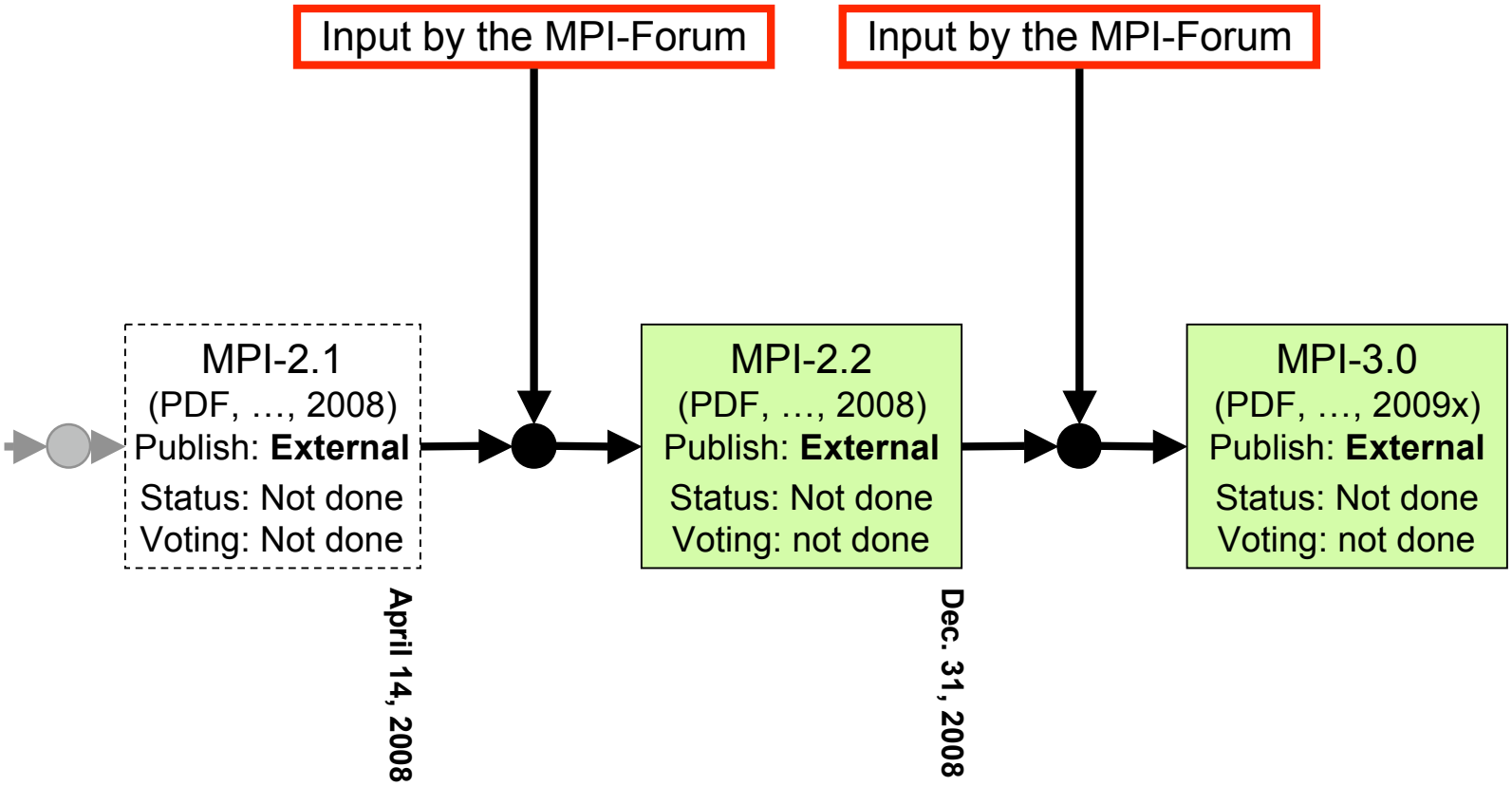
Is it okay?



MPI Standards document plan



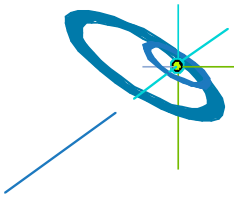
MPI Standards document plan



MPI 1.2.1 or MPI 1.3

- Should we name it MPI 1.3 instead of 1.2.1, including the change in MPI_GET_VERSION to MPI 1.3?
  - Yes: all-11
  - No: 2
  - Abstain: 9

- The rest of the document plan is okay?
  - Yes: all
  - No: 0
  - Abstain: 0



Ballot 3 –
1. MPI_COMM_PARENT instead of MPI_COMM_GET_PARENT

Question:

Do you accept this entry?

Yes:

All-2

No:

0

Abstain:

2

Mail discussion, proposed by Bill Gropp and Rusty Lusk, Mar 18, 2004

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/commparent/

MPI-2, page 179, lines 4-5 change

> **Thus, the names of MPI_COMM_WORLD, MPI_COMM_SELF, and MPI_COMM_PARENT will have the default of MPI_COMM_WORLD, MPI_COMM_SELF, and MPI_COMM_PARENT.**

to

> **Thus, the names of MPI_COMM_WORLD, MPI_COMM_SELF, and the communicator returned by MPI_COMM_GET_PARENT (if not MPI_COMM_NULL) will have the default of MPI_COMM_WORLD, MPI_COMM_SELF, and MPI_COMM_PARENT.**

MPI-2, page 94, line 3-5, change

> *** The manager is represented as the process with rank 0 in (the remote**
> *** group of) MPI_COMM_PARENT. If the workers need to communicate among**
> *** themselves, they can use MPI_COMM_WORLD.**

to

> *** The manager is represented as the process with rank 0 in (the remote**
> *** group of) the parent communicator. If the workers need to communicate**
> *** among themselves, they can use MPI_COMM_WORLD.**

Reason: MPI_COMM_PARENT is used where the communicator returned by MPI_COMM_GET_PARENT is meant. This reflects, I believe, an earlier version of the parent where we had a MPI_COMM_PARENT similar to MPI_COMM_WORLD.

Ballot 3 –
2. MPI_UNPACK_EXTERNAL

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

Mail discussion, proposed by Hubert Ritzdorf, May 09, 2001

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/unpackext/

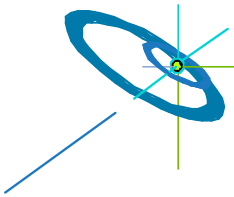
MPI-2, page 79, line 11 is

MPI_UNPACK_EXTERNAL (datarep, inbuf, incount, datatype, outbuf, outsize, position)

but should be

MPI_UNPACK_EXTERNAL (datarep, inbuf, insize, position, outbuf, outcount, datatype)

Reason: Wrong and inconsistent with rest of the definition of MPI_UNPACK_EXTERNAL.



Ballot 3 –
3. Additional C++ binding errors

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

Mail discussion proposed by Rolf Rabenseifner, Jul 17, 2001

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/CxxBindings/

MPI-2, page 337, line 31-32 reads

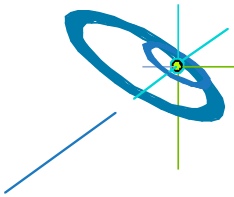
bool MPI::Win::Get_attr(const MPI::Win&win, int win_keyval, void* attribute_val) const

but should read

bool MPI::Win::Get_attr(int win_keyval, void* attribute_val) const

---

Reason: same as adopted correction in Ballot 1&2 → MPI 2.0 page 204, line 30



Ballot 3 –
4. MPI_REQUEST_CANCEL used where MPI_CANCEL intended

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

Mail discussion, proposed by Jeff Squyres and Rajeev Thakur, Oct. 31, 2006

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/req-cancel/

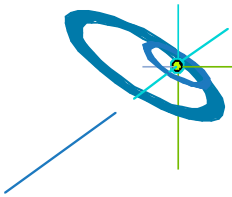
On page 172, line 37 in section 8.2, change

MPI_REQUEST_CANCEL

To

MPI_CANCEL

Reason: Typo



Ballot 3 – 5. Intercommunicator collective and datatypes

→ **Moved into Ballot 4**

Still TODO !

Question:

Do you accept this entry?

Yes:

No:

Abstain:

Mail discussion, proposed by Bill Gropp, Feb 25, 2000, **modified Jan 14, 2008**

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/iccoll/

MPI-2, page 162, line 47-48 reads (in MPI_ALLREDUCE)

Both groups should provide the same count value.

but should read

....

We may counter-check with following MPI 1.2 text whether the proposed new text is okay.

Still TODO !

Ballot 3 – 5. Intercommunicator collective and datatypes (continued)

→ **Moved into Ballot 4**

Still TODO !

Question:

Is the merging decision for MPI-2 Sect.3.2.7 okay?

Yes:

No:

Abstain:

- **Background is the MPI-1.2 text on MPI_Reduce datatype/count usage**

Blue: MPI 1.1, page 114, lines 1,28-30 Purple: MPI-2.0, page 26, lines 22-34

The routine is called by all group members using the **same** arguments for **count**, **datatype**, op, root and comm.

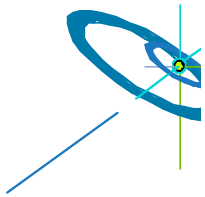
Bold font highlighting from me

…

The datatype argument of MPI_REDUCE must be compatible with op. Predefined operators work only with the MPI types listed in Section 4.9.2 and Section 4.9.3. **Furthermore, the datatype and op given for predefined operators must be the same on all processes**..

Note that it is possible for users to supply different user-defined operations to MPI_REDUCE in each process. MPI does not define which operations are used on which operands in this case. User-defined operators may operate on general, derived datatypes. In this case, each argument that the reduce operation is applied to is one element described by such a datatype, which may contain several basic values. **This is further explained in Section 4.9.4**

> *Advice to users.* Users should make no assumptions about how MPI_REDUCE is implemented. Safest is to ensure that the same function is passed to MPI_REDUCE by each process. *(Advice to users.)*



Overlapping datatypes are permitted in ``send'' buffers. Overlapping datatypes in ``receive'' buffers are erroneous and may give unpredictable results.

Ballot 3 – 5. Intercommunicator collective and datatypes (continued)

Moved into Ballot 4

Still TODO !

Question:

Do you accept this entry?

Yes:

No:

Abstain:

Mail discussion, proposed by Bill Gropp, Feb 25, 2000, **modified Jan 14, 2008**

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/iccoll/

MPI-2, page 163, line 22-24 reads (in MPI_REDUCE_SCATTER)

> **Within each group, all processes provide the same recvcounts argument, and the sum of the recvcounts entries should be the same for the two groups.**

but should read

> **Within each group, all processes provide the same type signature as defined by the recvcounts and datatype arguments, and the recvcounts entries and datatype should specify the same type signature for the two groups.**

Reason: Several of the intercommunicator collective operations contain statements along the lines of "Both groups should provide the same count value". However, what is really required is that the (count,datatype) tuples describe the same type signature. See MPI_Allreduce and MPI_Reduce_scatter. I propose a clarification that replaces the text that refers only to count to "Both groups should provide count and datatype arguments that specify the same type signature."

Ballot 3 –
6. const in C++ specification of predefined MPI objects

Question:

Do you accept this entry?

Yes:

All-5

No:

0

Abstain:

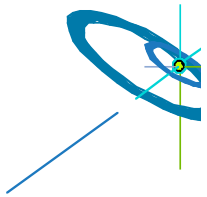
5

Mail discussion, by Richard Treumann and Rolf Rabenseifner, Jun 13 – Jul 26, 2001

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/cxxconstdtype/

- MPI-2, page 345, line 37: **Remove** the const from **const MPI::Op**.
- MPI-2, page 346, line 20: **Remove** the const from **const MPI::Group**.
- MPI-2, page 346, add after line 34:
  Advice to implementors: If an implementation does not change the value of predefined handles while execution of MPI_Init, the implementation is free to define the predefined operation handles as const MPI::Op and the predefined group handle MPI::GROUP_EMPTY as const MPI::Group. Other predefined handles must not be "const" because they are allowed as INOUT argument in the MPI_COMM_SET_NAME/ATTR and MPI_TYPE_SET_NAME/ATTR routines. (End of advice to implementors.)

- Reason: MPI_Init may change the predefined handles, because MPI 1.1, page 10, lines 9-10 says: "Opaque objects accessed by constant handles are defined and do not change value between MPI initialization (MPI_INIT() call) and MPI completion (MPI_FINALIZE() call)." Therefore they must not be defined as const in the MPI standard.
  I would allow one exception: The predefined ...._NULL handles, because as fare as I know, all implementations handle ..._NULL as (zero) constant of arbitrary datatype. See MPI-2, page 346, lines 4, 10, 12, 14, 16 (const in Ballot 1&2).



Ballot 3 –
7. Error in MPI_Scan Example

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

Mail discussion, proposed by A. Ayhan Kanmaz, May 14, 2003

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/scanexample/

MPI 1.1, page 128, line 11, in MPI-1.1 has an extraneous root argument.
That line should be

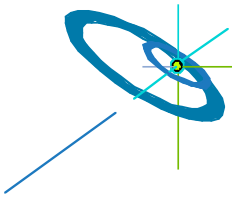
MPI_Scan(a, answer, 1, sspair, myOp, comm);

(instead of

MPI_Scan(a, answer, 1, sspair, myOp, root, comm);

)

Reason: MPI_Scan hasn‘t a root argument.



Ballot 3 –
8. Missing newline in Fortran binding

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

MPI-2, page 223, line 19. Change

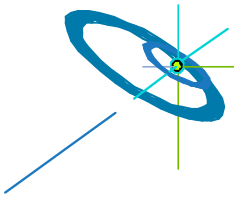
MPI_FILE_GET_VIEW(FH, DISP, ETYPE, FILETYPE, DATAREP, IERROR)
INTEGER FH, ETYPE, FILETYPE, IERROR
CHARACTER*(*) DATAREP, INTEGER(KIND=MPI_OFFSET_KIND) DISP

to

MPI_FILE_GET_VIEW(FH, DISP, ETYPE, FILETYPE, DATAREP, IERROR)
INTEGER FH, ETYPE, FILETYPE, IERROR
CHARACTER*(*) DATAREP
INTEGER(KIND=MPI_OFFSET_KIND) DISP

in io-2.tex. (Replace the comma after the declaration of datarep)

Reason: Formatting error



Ballot 3 –
9. Misspelled argument in Fortran binding

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

MPI-2, page 66, line 26, change

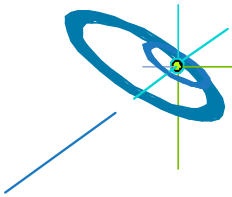
MPI_TYPE_CREATE_HVECTOR(COUNT, BLOCKLENGTH, STIDE, OLDTYPE, NEWTYPE, IERROR)
INTEGER COUNT, BLOCKLENGTH, OLDTYPE, NEWTYPE, IERROR
INTEGER(KIND=MPI_ADDRESS_KIND) STRIDE

to

MPI_TYPE_CREATE_HVECTOR(COUNT, BLOCKLENGTH, STRIDE, OLDTYPE, NEWTYPE, IERROR)
INTEGER COUNT, BLOCKLENGTH, OLDTYPE, NEWTYPE, IERROR
INTEGER(KIND=MPI_ADDRESS_KIND) STRIDE

in misc-2.tex (Replace STIDE with STRIDE).

Reason: Typo



Ballot 3 –
10. Error in MPI-1, Example 3.12

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

Proposed by NN, and Bill Gropp, Jan. 3, 2008

Mail discussion: Examples in Chapter 3 of MPI 1.1 require several fixes.

MPI 1.1, Example 3.12, page 43, line 47 and page 44, lines 1, 5, 8, 10, and 13, the communicator argument **comm** must be added before the req argument.

Mail discussion: The **ierr** argument must be added at the end of the argument list in the calls to MPI_COMM_RANK and MPI_WAIT in MPI 1.1, page 43, line 43, and page 44, lines 6 and 14.

Mail discussion: The **ierr** argument must be added at the end of the argument list in the calls to MPI_WAIT in MPI 1.1, page 44, lines 35 and 36.

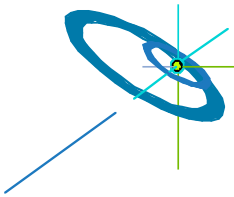
Mail discussion: The lines in MPI 1.1, page 52, line 45, and page 53, line 17

IF (status(MPI_SOURCE) = 0) THEN

should be

IF (status(MPI_SOURCE) .EQ. 0) THEN

Reasons: Obvious / Syntax error



Ballot 3 –
11. Error in MPI-1, Example 3.34

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

Mail discussion, proposed by Bettina Krammer

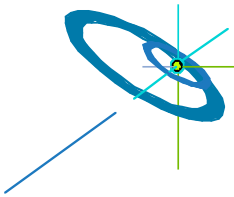
http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/ex334/

MPI 1.1, page 80, line 2,

The variable base should be declared as **MPI_Aint**, not **int**, in Example 3.34.

Reason:

The variable base (declared on this line) is used to store the address output from MPI_Address. On systems with addresses longer than 32 bit, a truncation will cause wrong execution of the program.



Ballot 3 –
12. Change MPI-2, page 343, lines 22-23

Mail discussion, proposed by Jeff Squyres, Nov. 27, 2007

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/constbottom/

Prelimi-nary question:

Do you accept Proposal 2 ?

1st vote on 1/14

Yes:

(All-6)

No:

(1)

Abstain:

(5)

Change MPI-2, page 343, lines 22-23

// Type: const void * MPI::BOTTOM

to (Proposal 1)

// Type: void * MPI::BOTTOM

to (Proposal 2)

// Type: void * const MPI::BOTTOM

Questions: (2nd votes on 1/16/2008)

Do we remove the const before void?

Yes: all-1 No: 0 Abstain: 1

Do we add the const before MPI::BOTTOM?

Yes: all-14 No: 1 Abstain: 13

Reason:

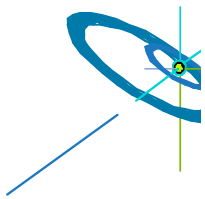
See mail discussion on next slides

Jeff Squyres + Alexander Supalov + Erez Haba are reviewing the topic:

e.g., const allows optimized allocation of the value in read-only pages

This declaration must reflect the rule defined in MPI 1.1, page 10, lines 7-11:

All named constants, with the exception of MPI_BOTTOM in Fortran, can be used in initialization expressions or assignments. These constants do not change values during execution. Opaque objects accessed by constant handles are defined and do not change value between MPI initialization (MPI_INIT() call) and MPI completion (MPI_FINALIZE() call).



Ballot 3 –
12. Change MPI-2, page 343, lines 22-23 (discussion)

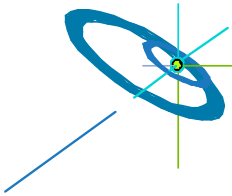
Jeff Squyres, Nov. 27, 2007

A user recently raised an issue that I just looked into and discovered a problem with the C++ binding for MPI::BOTTOM. In the spec, MPI::BOTTOM is defined to be of type (const void*). However, all receive buffers are defined to be of type (void*) -- such as for the various flavors of point-to-point receive, the receive buffer for collectives, etc. This means that you'll get a compiler error when trying to use MPI::BOTTOM as a receive buffer:

bottom.cc:81: error: invalid conversion from const void*' to void*'

bottom.cc:81: error: initializing argument 1 of virtual void MPI::Comm::Bcast(void*, int, const MPI::Datatype&, int) const'

A user can cast away the const-ness of MPI::BOTTOM, but that seems inelegant/wrong. I don't yet have a solution to this problem; I raise it here so that it gets added to the list of issues to be addressed in MPI-2.1.



Ballot 3 –
12. Change MPI-2, page 343, lines 22-23 (discussion)

Dave Goodell, Nov. 27, 2007

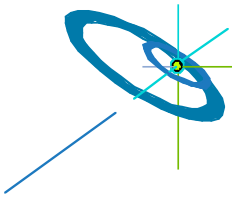
Looks like the const is on the wrong side of the declaration. That is, unless my C++ is too rusty it should instead be something like:

```
namespace MPI {
    ...
    extern void * const BOTTOM;
    ...
}
```

"const TYPE * FOO" indicates that the data pointed to by FOO is read-only. So "*FOO = BAR;" would be an illegal statement.

"TYPE * const FOO" indicates that the memory holding the value of FOO is read-only. So "FOO = &BAR;" would be an illegal statement.

The latter seems to be what is desired for MPI::BOTTOM: an address that cannot be changed but a the data that it references can.

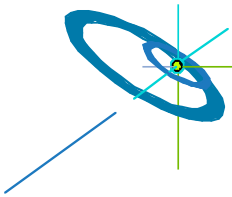


Ballot 3 –
12. Change MPI-2, page 343, lines 22-23 (discuss., cont'd)

Jeff Squyres, Nov. 28, 2007

> Good point. I think you're right -- I ran a few tests to convince myself that changing the type of MPI::BOTTOM to (void * const) won't break anything in terms of the other existing bindings.
>
> However, in terms of what MPI::BOTTOM *should* be, shouldn't it be *both* consts? We don't want the value to change, nor do we want the pointed-to-contents where it points to change:
>
> extern const void * const BOTTOM;
>
> Technically, though, with your suggestion, you couldn't change the pointed-to contents without casting anyway (because you can't assign to *(void*)). So this might be a good enough solution.

My opinion: Dave Goodell is right. → Therefore back to the proposel-slide.



Ballot 3 –
13. MPI 1.1, strlen in first pt-to-pt example

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

Mail discussion, proposed by Bill Gropp, Jan 2, 2008

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/strlen/

In MPI 1.1, page 16, line 23, use

strlen(message) + 1

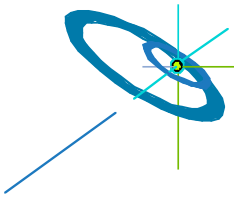
instead of

strlen(message)

in the MPI_Send call.

Reason:

In the MPI-1 document, on page 16 (first page of chapter 3), the example uses strlen(message) for the number of characters in the string message to send, and then uses printf to print that message when received. This fails to send the trailing null, so in the MPI_Send call, the length should be strlen(message) + 1 on line 33.



Ballot 3 –
14. Formatting error on MPI 1.1, page 58

Question:

Do you accept this entry?

Yes:

all

No:

0

Abstain:

0

Mail discussion, proposed by Bill Gropp, Jan 3, 2008

http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/discuss/persistypo/

A LaTeX line break is needed in MPI 1.1, page 58, line 44, in Section 3.9.

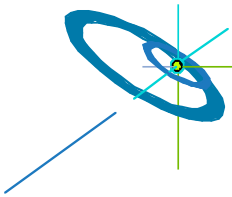
The text should read

be invoked in a sequence of the form,

Create (Start Complete)* Free

where * indicates zero or more repetitions. If the same communication ...

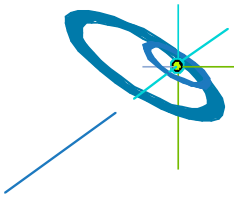
Reason: Formatting error



Ballot 4

- Requirement:
  - A final text must be available at March meeting
- Everybody is invited to check
  - his/her own proposals
  - or proposals where he/she was involved in the discussion

  at
  - http://www.cs.uiuc.edu/homes/wgropp/projects/parallel/MPI/mpi-errata/
  - And add a final text proposal
    - **Identifying exactly MPI 1.1 / MPI-2, page, and lines**
    - **That should be modified or added**
- Currently, nearly no proposal has a final text
- MPI 2.1: only clarifications and errata
- Deadline for MPI 2.1: a week before March meeting
  - All other → Bill Gropp 2.2
  - some from Bill's current 2.2 list → may come to 2.1

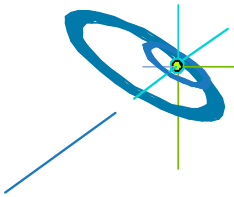


Change-Log as Annex

- Who wants in MPI 2.1 an Annex telling effective (user and implementors visible) mods between MPI 2.0 and MPI 2.1?
- Yes: all=43
- No: 0
- Abstain: 0

- Should we keep old such histories in future standards?
- Yes: 8
- No: 13
- Abstain: 43-8-13=22

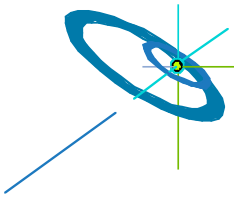
- Changbars for all in the final combined document?
- Yes: 1
- No: 28
- Abstain: 43-28-1=14



MPI 1.3 combined document

- MPI 1.1 **+** Chap. 3 of MPI-2 (Version 1.2 of MPI) + some errata will be combined to
  → **MPI 1.3 combined document**

  - Jan.08 meeting:
    Short discussion and **defining a review group** who is reviewing
    the **MPI 1.3 merging plan** (printed copies available)
    and the **MPI 1.3 combined document**
  - See e-mail: From: Rainer Keller, Subject: Re: [mpi-21] Documents
    Date: Mon, 7 Jan 2008 12:13:14 +0100
  - Reporting by e-mail on mpi-21 reflector
  - Corrections if necessary
  - Final report of the reviewers at March 2008 meeting
  - **1st vote** by the MPI Forum at April 2008 meeting
  - **2nd (final) vote** by the MPI Forum at June 2008 meeting



MPI 1.3 combined document

Question:

Should we include MPI-2-chapter3 +all new MPI-1 errata into the combined MPI 1.3?

Yes:

All-1

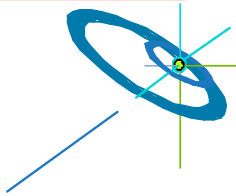
No:

0

Abstain:

1

- Do we want to include the MPI 1.1 errata already into this MPI 1.3 document?
- Pro:
  - This document is a “final” document telling the MPI-1 standard.
- Con:
  - Formally, it is not the right place. New stuff must be in MPI 2.1.
- My recommendation:
  - The “pro” outweighs the “con”.



MPI 1.3 combined document – the "merging document"

Question:

Is the new history text okay?

Yes:

all

No:

0

Abstain:

0

Merge of MPI-1.1 (June 1995) and MPI-1.2 (July 1997) plus new Errata (MPI 1.2.1, 2008)

Versions-History page:

Version 1.3: ?????, 2008. This document combines the previous documents MPI 1.1 (June 12, 1995) and the MPI 1.2 Chapter in MPI-2 (July 18, 1997). Additional errata collected by the MPI Forum referring to MPI 1.1 and MPI 1.2 are also included in this document.

Version 1.2: July, 18 1997. The MPI-2 Forum introduced MPI 1.2 as Chap.3 in the standard "MPI-2: Extensions to the Message-Passing Interface", July 18, 1997." This section contains clarifications and minor corrections to Version 1.1 of the MPI Standard. The only new function in MPI-1.2 is one for identifying to which version of the MPI Standard the implementation conforms. There are small differences between MPI-1 and MPI-1.1. There are very few differences ~~(only those discussed in this chapter)~~ between MPI-1.1 and MPI-1.2, but large differences ~~(the rest of this document)~~ between MPI-1.2 and MPI-2.

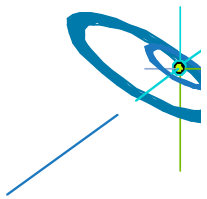
Version 1.1: June, 1995. Beginning in March, 1995, the Mes…

Version 1.0: June, 1994. The Message Passing Interface Forum (MPIF), with participation from over 40 organizations, …

New text

This text is from MPI 2.0, page 21, lines 14-19, but parentheses removed

Existing MPI 1.1 text

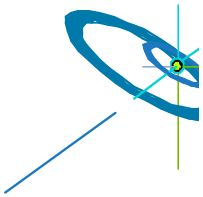


MPI 1.2 combined document – the “merging document”

- 3.1: Integrated MPI_Get_version into Environmental Section, Inquiries -- from MPI-2, p. 21
  (changes to appLang.tex and inquiry.tex)
  * the section title in MPI-2 is "Version Number", should not be changed?
  * MPI-2.0 Sect. 3.1 page 21 line 21 - page 22 line 2
    added as new Sect. 7.1.1 in MPI-1.1
    before current MPI-1.1 Sect 7.1.1 on page 190 line 21
    remove last sentence on MPI-2.0 page 22 line 2:
    "Its C++ binding can be found in the Annex, Section B.11."

- 3.2: MPI-1.0 and MPI-1.1 Clarifications
  * MPI-2.0 page 22 lines 4-10 not used (removed)

- 3.2.1: MPI_INITIALIZED: -- from MPI-2, p. 21 lines 14-15
  * added in MPI-1.1 page 200 line 11.
  * MPI-1.1 page 200 lines 10-11 must be modified because MPI_GET_VERSION
  * maybe also called before MPI_Init (And MPI_FINALIZED in MPI-2.0):
  Changed: "is the only function that may be called before" to
  "It is one of the few routines that “



- 3.2.2: Include clarification of MPI_FINALIZE -- from MPI-2, p. 22 line 18 - p. 24 line 48:
  Replaces MPI-1.1 paragraph page 199 lines 46-48

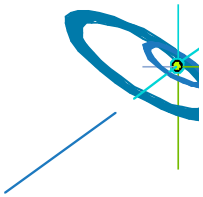
MPI 1.2 combined document – the “merging document”

- 3.2.3 Clarification of status after MPI_WAIT and MPI_TEST -- from MPI-2, p. 25 lines 2-12
  Position in standard not completely obvious.
  Fits best after the definition of empty statuses in MPI-1, 3.7.3
  * i.e., after MPI-1.1 page 41 line 20

- 3.2.4 Clarification of MPI_INTERCOMM_CREATE -- from MPI-2, p. 25.
  Added to the section on Inter-Communication
  * Delete the text in parenthesis on MPI-1.1 page 158 line 31.
  * Substitute the sentence MPI-1.1 page 155 lines 36-37
    by MPI-2.0 page 25 lines 37-47

- 3.2.5 Clarification of MPI_INTERCOMM_MERGE -- from MPI-2, p. 26 lines 2-4
  Added paragraph on errorhandlers to MPI_INTERCOMM_MERGE
  * after MPI-1.1 page 160 line 13

- 3.2.6 Clarification of MPI_TYPE_SIZE -- from MPI-2, p. 26 lines 11-13
  Added advice to users
  * after MPI-1.1 page 70 line 43



MPI 1.2 combined document – the “merging document”

Question:

Is the merging decision for MPI-2 Sect.3.2.7 okay?

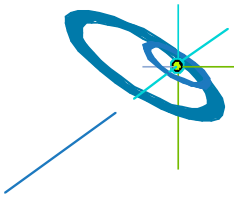
See next slide!

- **3.2.7 Clarification of MPI_REDUCE -- from MPI-2, p. 26**

  Required extensive modification:

  * **MPI-2.0 page 26 lines 22-28 is substituting the text on MPI-1.1 page 114 lines 25-26.**
  * **The sentence MPI 1.1, page 114, lines 26-27 “User-defined operators may operate on general, derived datatypes.” is not removed.**
  * **MPI-2.0 page 26 lines 29-35 must be added after MPI-1.1 page 114 line 30.**
  * **No need for additional new text "This is further explained in Section 4.9.4"**

Review of this proposal on the next slide.



Question:

Is the merging decision for MPI-2 Sect.3.2.7 okay?

Yes:

all

No:

0

Abstain:

0

MPI 1.2 combined document – the "merging document" New proposal on Jan 2008 meeting

- **3.2.7 Clarification of MPI_REDUCE -- from MPI-2, p. 26**

Blue: MPI 1.1, page 114, lines 25-30 Purple: MPI-2.0, page 26, lines 22-34

~~The datatype argument of MPI_REDUCE must be compatible with op. Predefined operators work only with the \MPI/ types listed in Sec. \ref{coll-predefined-op} and Sec. \ref{coll-minloc-maxloc}.~~ The datatype argument of MPI_REDUCE must be compatible with op. Predefined operators work only with the MPI types listed in Section ~~4.9.2~~ \ref{coll-predefined-op} and Section ~~4.9.3~~ \ref{coll-minloc-maxloc}. Furthermore, the datatype and op given for predefined operators must be the same on all processes.

Note that it is possible for users to supply different user-defined operations to MPI_REDUCE in each process. MPI does not define which operations are used on which operands in this case. **User-defined operators may operate on general, derived datatypes**. In this case, each argument that the reduce operation is applied to is one element described by such a datatype, which may contain several basic values. This is further explained in Section~\ref{subsec:coll-user-ops}.

> *Advice to users.* Users should make no assumptions about how MPI_REDUCE is implemented. Safest is to ensure that the same function is passed to MPI_REDUCE by each process. *(Advice to users.)*

Overlapping datatypes are permitted in ``send'' buffers. Overlapping datatypes in ``receive'' buffers are erroneous and may give unpredictable results.

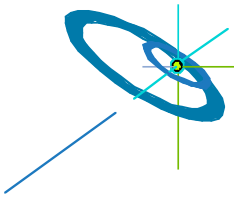
This sentence is kept although MPI-2 requires deleting. (The content is correct)

MPI 1.2 combined document – the “merging document”

- 3.2.8 Clarification of Error Behaviour of Attribute Callback Function -- from MPI-2, p. 26 lines 38-39

  Added to section 5.7.1, right after definition of delete_fn

  * i.e., after MPI-1.1 page 170 line 7



MPI 1.2 combined document – the “merging document”

Question:

Should we keep the rationale MPI-2 Sect.3.2.9 page 27 line 1-32?

See next slides.

- **3.2.9 Clarification of MPI_PROBE and MPI_IPROBE -- from MPI-2, p. 27**

  Replaced text, left out rationale...

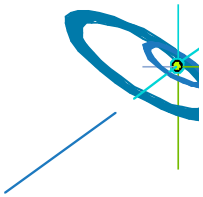
  *** The rationale may be kept, but all references should be**

  *** referencing the MPI 1.1 document (and not the new combined document)**

  *** TODO: Decision on Rationale must be done by MPI-2.1 Forum.**

  The location for the rationale would be directly after the paragraph with the substituted text, i.e., after MPI 1.1, page 52, line 4.

Decision by the MPI-Forum (on next slides): The rationale is removed.



MPI 1.2 combined document – the "merging document"

- **3.2.9 Clarification of MPI_PROBE and MPI_IPROBE -- from MPI-2, p. 27**

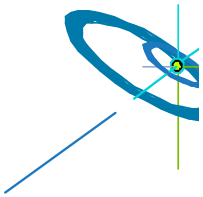
Page 52, lines 1 thru 3 (of MPI 1.1, the June 12, 1995 version without changebars)

> **A subsequent receive executed with the same context, and the source and tag returned in status by MPI_IPROBE will receive the message that was matched by the probe, if no other intervening receive occurs after the probe. If the receiving process is multi-threaded, it is the user's responsibility to ensure that the last condition holds.**

become:

> **A subsequent receive executed with the same communicator, and the source and tag returned in status by MPI_IPROBE will receive the message that was matched by the probe, if no other intervening receive occurs after the probe, and the send is not successfully cancelled before the receive. If the receiving process is multi-threaded, it is the user's responsibility to ensure that the last condition holds.**

> ***Rationale.***
>
> **The following program shows that the original MPI-1.1 definitions of cancel and probe are in conflict:**



MPI 1.2 combined document – the “merging document”

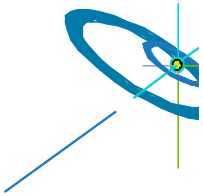
- 3.2.9 Clarification of MPI_PROBE and MPI_IPROBE -- from MPI-2, p. 27

Rationale.

The following program shows that the original MPI-1.1 definitions of cancel and probe are in conflict:

```
Process 0                    Process 1
---------                    ---------
MPI_Init();                  MPI_Init();
MPI_Isend(dest=1);

                             MPI_Probe();
MPI_Barrier();               MPI_Barrier();
MPI_Cancel();
MPI_Wait();
MPI_Test_cancelled();
MPI_Barrier();               MPI_Barrier();
                             MPI_Recv();
```



MPI 1.2 combined document – the "merging document"

Question:

Should we keep the rationale MPI-2 Sect.3.2.9 page 27 line 1-32?

Yes:

0

No:

All-7

Abstain:

7

- 3.2.9 Clarification of MPI_PROBE and MPI_IPROBE -- from MPI-2, p. 27

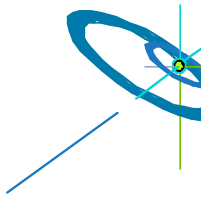
Since the send has been cancelled by process 0, the wait must be local (MPI 1.1, page 54, line 13) and must return before the matching receive. For the wait to be local, the send must be successfully cancelled, and therefore must not match the receive in process 1 (MPI 1.1, page 54 line 29).

However, it is clear that the probe on process 1 must eventually detect an incoming message. MPI 1.1, page 52 line 1 makes it clear that the subsequent receive by process 1 must return the probed message.

The above are clearly contradictory, and therefore the text "…and the send is not successfully cancelled before the receive" must be added to MPI 1.1, line 3 of page 54.

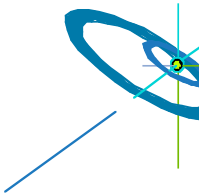
An alternative solution (rejected) would be to change the semantics of cancel so that the call is not local if the message has been probed. This adds complexity to implementations, and adds a new concept of "state" to a message (probed or not). It would, however, preserve the feature that a blocking receive after a probe is local.

(End of rationale.)



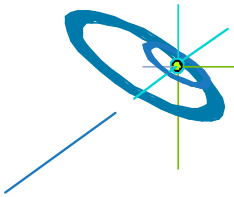
MPI 1.2.1 combined document – Review Group

- The review group has to check the merging locations shown in the "merging document" from Rainer Keller
- They have to check the final "combined document", whether it implements the decisions in the "merging document"
- Proposal:
  - At least 4 persons to check the "merging document" and the final combined document based on the decisions in the merging document
  - MPI 1.3 reviewing group:
    1. **Bill Gropp** (@meeting Jan-2008)
    2. **Rolf Rabenseifner** (@meeting Jan-2008)
    3. **Adam Moody**
    4. **Puri Bangalore**
    5. **Terry Dontje** (not @meeting Jan-2008)
    6. **William Yu** (not @meeting Jan-2008)



MPI 2.0 combined document

- There are 11 topics that must be decided or confirmed by the MPI Forum → next slides
- For further details we need a review group → slide at the end



MPI 2.0 merging decisions

Question:

The MPI 2.0 combined document title-page should be as stated here?

Yes:

All=41

No:

0

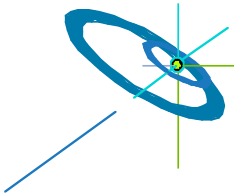
Abstain:

0

1.) The title of the combined document:

MPI: A Message-Passing Interface Standard

Version 2.1



MPI 2.0 merging decisions

Question:

The MPI 2.0 combined document title-page should be as stated here in 2.+3.?

Yes:

all

No:

0

Abstain:

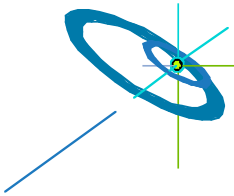
0

2.) The date of the merged document is fixed when it is released (in 2008).

3.) Ackno on the title page:

> **"This work was supported in part by ARPA, NSF and DARPA under grant ASC-9310330, the National Science Foundation Science and Technology Center Cooperative Agreement No. CCR-8809615, and the NSF contract CDA-9115428, and by the Commission of the European Community through Esprit project P6643 and under project HPC Standards (21111)."**

4.) Do we add on 2.1 already new supporters? Yes – offline per e-mail



MPI 2.0 merging decisions

Question:

The MPI 2.0 combined document abstract should be as stated here?

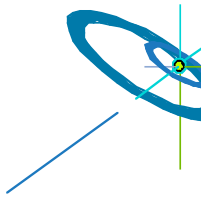
Yes:

No:

Abstain:

4.) Abstract

"This document describes the MPI standard version 2.1 in one combined document. This document combines the content from the previous standards "*MPI: A Message-Passing Interface Standard, June 12, 1995*" **(MPI-1.1) and** *"MPI-2: Extensions to the Message-Passing Interface, July, 1997"* **(MPI-1.2 and MPI-2.0). The standard MPI-1.1 includes point-to-point message passing, collective communications, group and communicator concepts, process topologies, environmental management, and a profiling interface. Language bindings for C and Fortran are defined. The MPI-1.2 part of the MPI-2 document contains clarifications and corrections to the MPI-1.1 standard and defines MPI-1.2. The MPI-2 part of the MPI-2 document describes additions to the MPI-1 standard and defines the MPI standard version 2.0. These include miscellaneous topics, process creation and management, one-sided communications, extended collective operations, external interfaces, I/O, and additional language bindings (C++). Additional clarifications and errata corrections are included."**



→ offline e-mail : be specific on errata doc. And include MPI 1.3

MPI 2.0 merging decisions

5.) Copyright years

1993, 1994, 1995, 1996, 1997, 2008

Question:

The MPI 2.0 combined document copyright years should be as stated here?

Yes:

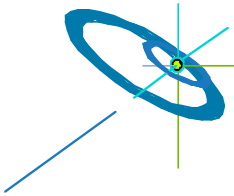
all

No:

0

Abstain:

0



MPI 2.0 merging decisions

Still TODO !

6.) New entries on the history page → **Offline per e-mail**

Version 2.1: <date>, 2008. This document combines the previous documents MPI 1.3 (????, 2008) and MPI-2.0 (July 18, 1997). Certain parts of MPI 2.0, such as some sections of Chapter 4, Miscellany, and Chapter 7, Extended Collective Operations have been merged into the Chapters of MPI 1.3. Additional errata and clarifications collected by the MPI Forum are also included in this document.

Version 1.3: <date>, 2008. This document combines the previous documents MPI 1.1 (June 12, 1995) and the MPI 1.2 Chapter in MPI-2 (July 18, 1997). Additional errata collected by the MPI Forum referring to MPI 1.1 and MPI 1.2 are also included in this document.

Version 2.0: <date>, 1997. Beginning after the release of MPI 1.1, the MPI Forum began meeting to consider corrections and extensions. MPI-2 has been focused on process creation and management, one-sided communications, extended collective communications, external interfaces and parallel I/O. A miscellany chapter discusses items that don't fit elsewhere, in particular language interoperability.“

Version 1.2: July, 18 1997. The MPI-2 Forum introduced MPI 1.2 as Chap.3 in the standard "MPI-2: Extensions to the Message-Passing Interface", July 18, 2007.“ …

Version 1.1: June, 1995. Beginning in March, 1995, the Message …

Version 1.0: June, 1994. The Message Passing Interface Forum …

As already voted for MPI 1.2.1

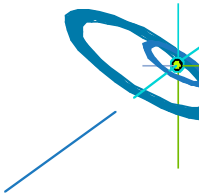
Question:

The MPI 2.0 combined document Versions-list should be as stated here?

Yes:

No:

Abstain:



MPI 2.0 merging decisions

Question:

The MPI 2.0 combined document Ackno-pages should be as stated here?

Yes:

all

No:

0

Abstain:

0

7.) The acknowledgment pages of both documents are printed one after the next.

The first list of persons is introduced with the following new line:

"Those who served as primary coordinators in MPI 1.0 and MPI 1.1 are:"

And the MPI-2 list of persons is introduced with:

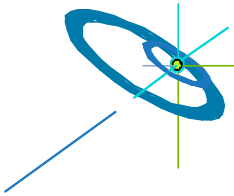
"Those who served as primary coordinators in MPI 1.2 and MPI 2.0 are:"

For the merging process I would add:

"The editors of the combined documents have been:

-- Rainer Keller (MPI 1.3 - combined document)

-- Rolf Rabenseifner (MPI 2.1 - combined document)"



MPI 2.0 merging decisions

8.) Before the definition of each deprecated interface, the following sentence is added:

"The following function is deprecated and is superseded by …… in MPI 2.0"

Question:

The MPI 2.0 combined document entries of depre-cated routines should be as stated here?

Yes:

all

No:

0

Abstain:

0

MPI 2.0 merging decisions

Question:

The MPI 2.0 combined document structure should be as stated here?

Yes:

All ?

No:

0 ?

Abstain:

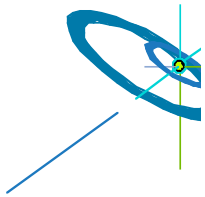
0 ?

9.) Sequence of all chapters:

- First all MPI-1, except Profiling
- All MPI-2, except Extended Collective (it is fully merged into corresponding MPI-1.1 chapters)
- Profiling Chapter
- Only one merged Appendix

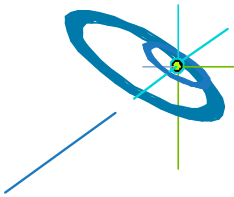
The following MPI-2.0 sections are merged into the corresponding MPI-1.1 sections:

- MPI-2.0 Chap. 4 Miscellany, except
  - -- MPI-2.0 Sect. 4.10 (The Info Object) (Move? Yes:no:abstain)
  - -- MPI-2.0 Sect. 4.11 (Memory Allocation) → MPI-1 Env. Manag. (5:2:?)
  - -- MPI-2.0 Sect. 4.12 (Language Interoper.) → End of Lang.Bind 10.3 (17:0:?)
  - -- MPI-2.0 Sect. 4.17 (Functions and Macros) → Terms & Conven. 2.6.5 (5:0:?)
- MPI-2.0 Chap. 5 Extended Collective Operations
- MPI-2.0 Sect. 8.8 New Attribute Cacching Functions
- MPI-2.0 Sect. 8.9 Duplicating a Datatype



Describe the chapter moving in a change-log

- Describe the chapter moving in a change-log?
  - Yes: all
  - No: 0
  - Abstain: 0



MPI 2.0 merging decisions

Question:

The MPI 2.0 combined document … should be as stated here?

Yes:

No:

Abstain:

Not decided by the Forum

10.) The new "Version Number" section was put at the beginning of the MPI-1.1 Chap. "Environmental Management" (done in MPI 1.2)

The Section "Portable MPI Process Startup" is put at the end of the MPI-1.1 Chapter "Environmental Management".

Okay?

Where to include Section „Memory Allocation“?

7 MPI Environmental Management
- 7.1 Implementation information
  - 7.1.1 Version number
  - 7.1.2 Environmental Interface
- 7.2 Error handling
- 7.3 Error codes and classes
- 7.4 Timers and synchronization
- 7.5 Startup
- 7.6 Portable MPI Process Startup

from MPI 1.2

from MPI 2.0

MPI 2.0 merging decisions

Question:

The MPI 2.0 combined document C++ interfaces should be handled as stated here?

Yes:

all

No:

0

Abstain:

0

11.) The MPI-1.2 C++ interfaces in MPI-2.0 App. B.3 - B.12 are written in a syntax that is different to the syntax used in all other C++ declarations.

For consistency, in B.3 - B.12, the

→ **this type of declaration should be used in Annex A**

"namespace MPI {

standard-type mpi-routine(standard-type arg1, mpi-type arg2...)

mpi-type mpi-routine(standard-type arg1, mpi-type arg2...)

}"

should be modified into

"standard-type MPI::mpi-routine(standard-type arg1, MPI::mpi-type arg2...)"

"MPI::mpi-type MPI::mpi-routine(standard-type arg1, MPI::mpi-type arg2...)„

Examples: static MPI::Intercomm MPI::Comm::Join(const int fd)

MPI::Intercomm MPI::Intercomm::Create(const Group& group) const

MPI::Intercomm MPI::Intracomm::Create(const Group& group) const

void MPI::Comm::Barrier() const = 0

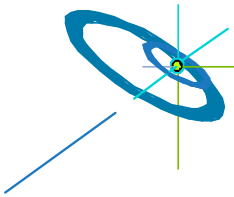
Straw votes okay for merging details?

With MPI 1.3 and MPI 2.1 documents:

- Is it okay to have only a final „reading“ (=review report) and two official votes, instead of already doing official votes on some details?
  Official (institutional) votes:
  - Yes:
  - No:
  - Abstain:

- Reason: The merging does not modify the standard. Only formatting and editorial wording is rarely modified.

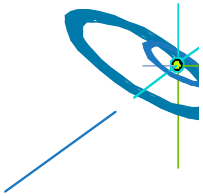
- **(This slide was skipped at January 2008 meeting.)**



MPI 2.0 Merging Review Group

We need reviewers for: (**bold=large**) Reviewers: (**next slide**)

- **Frontmatter** (**large**)
- **Chapter 1: Introduction to MPI** (**large**)
- Chapter 2: MPI-2 Terms and Conventions (simple)
- **Chapter 3: Point-to-Point Communication** (**large**)
- **Chapter 4: Collective Communication** (**large**)
- **Chapter 5: Groups, Context, and Communicators** (**large**)
- Chapter 6: Process Topologies (simple)
- **Chapter 7: MPI Environmental Management** (**large**)
- Chapter 8: Miscellany (simple)
- Chapter 9: Process Creation and Management (simple)
- Chapter 10: One-Sided Communication (simple)
- Chapter 11: External Interfaces (simple)
- Chapter 12: I/O (simple)
- **Chapter 13: Language Bindings** **(large)**
- Chapter 14: Profiling Interface (simple)
- Bibliography (simple)
- **Annex A** (**large**)



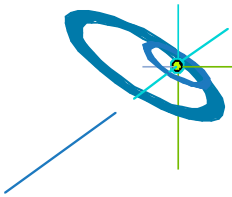
We need reviewers for: (**bold=large**)	Reviewers: (**green=@meeting**)
Frontmatter	**Rusty Lusk, Bill Gropp**
Chap. 1: Introduction to MPI	**Rusty Lusk, Bill Gropp, Karl Feind, Adam Moody**
Chap. 2: MPI-2 Terms and Conventions	Tony Skjellum, **Bill Gropp, Richard Barrett**
Chap. 3: Point-to-Point Communication (incl. sections from MPI-2 Misc. + 8.9)	**Rich Graham**, Jespar Larsson Traeff, **George Bosilca**, **Steve Poole, Kannan Narasimhan**, David Solt, **B. Gropp Matt Koop**
Chap. 4: Collective Communication (incl. sections from MPI-2 Ext. Collect.)	Steven Ericsson-Zenith, **Edgar Gabriel, Rajeev Thakur, Bill Gropp, Adam Moody, Georg Bosilca**
Chap. 5: Groups, Context, and Communicators (incl. sections from MPI-2 Ext.Col. + 8.8)	Steven Ericsson-Zenith, **Edgar Gabriel, Bill Gropp, Georg Bosilca, Robert Blackmore**
Chap. 6: Process Topologies	**Rusty Lusk, Bill Gropp, Richard Barrett**
Chap. 7: MPI Environmental Management (incl. sections from MPI-2 Misc.)	**Rich Graham**, Jespar Larsson Traeff, **George Bosilca**, **Steve Poole, Kannan Narasimhan**, David Solt, **B. Gropp**
Chap. 8: Miscellany	**Rich Graham**, **George Bosilca**, **Steve Poole, Kannan Narasimhan**, **B. Gropp**
Chap. 9: Process Creation and Management	Dries Kimpe, **Rusty Lusk, Georg Bosilca, Bill Gropp, Kalem Karian**
Chap. 10: One-Sided Communication	Ericsson-Zenith, Jespar Larsson Traeff, Martin Schulz, **Bill Gropp, Darius Buntinas**
Chap. 11: External Interfaces	**Bronis de Supinski, Bill Gropp**
Chap. 12: I/O	**Rajeev Thakur**, Joachim Worringen, **Bill Gropp**
Chap. 13: Language Bindings	**Jeff Squyres, Steve Poole, Purushotham Bangalore, Bill Gropp, Erez Haba, Alexander Supalov**
Chap. 14: Profiling Interface	**Bronis de Supinski, Bill Gropp, Jeff Brown**
Bibliography	**Rusty Lusk, Bill Gropp**
Annex A	**Jeff Squyres, Steve Poole, Purushotham Bangalore, Bill Gropp, Alexander Supalov**

MPI 2.1 - Telecon

- I'll not be available on Feb. 11-13 (scheduled dates)
- There is no need because all Ballot 4 work will be done by e-mail
- And MPI 1.3, MPI 2.1 merging and Ballot 3 is done.

- → No telecon for the MPI 2.1 chapter
- Okay?

- (Accepted by the Forum due to the reasons mentioned above)



MPI 2.1

- Thank you very much for your strong support on MPI 2.1!