MPI 3.0 Survey Results

Josh Hursey (Indiana University)
Jeff Squyres (Cisco Systems, Inc.)

Overview

‣ Disclaimer

‣ Review of Survey Questions

‣ Obvious Results from Atlanta Meeting

‣ Unanswerable Questions from Data

‣ Deeper Analysis

‣ Questions for Forum

Disclaimer

Disclaimer

▶ We are not statisticians.

- ▶ We tried to represent the data in an unbiased manner as possible when attempting to answer the questions posed by the MPI Forum.

▶ Some of the Forum's questions were modified to turn them into forms that can be answered from the data.

▶ The results of the survey are meant to be taken as additional input to the MPI Forum members from the broader MPI community.

- ▶ The MPI Forum will decide how significantly they incorporate these responses into the decision making process on a per-response basis.

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE

Review of Survey Questions

Demographics

Did you attend the MPI Forum BOF at SC09?

Yes
No

Demographics

Which of the following best describes you?

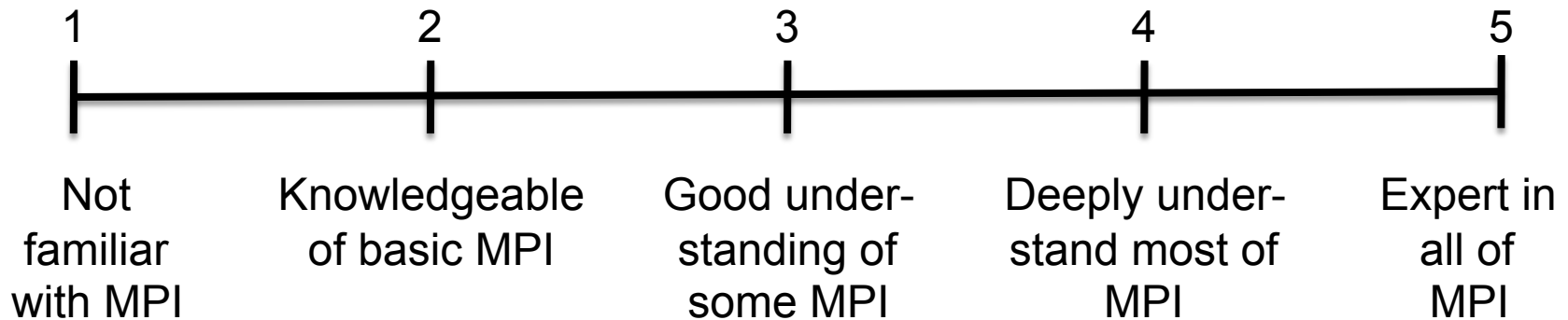| |
|---|
| User of MPI applications |
| MPI application developer |
| Library / middleware developer (that uses MPI) |
| MPI implementer |
| Academic educator / researcher |
| Student |
| Project /program / general management |
| Other |

Other (text)

Demographics

Rate your expertise with the MPI standard.

1	2	3	4	5
Not familiar with MPI	Knowledgeable of basic MPI	Good under-standing of some MPI	Deeply under-stand most of MPI	Expert in all of MPI

Apps

Think of an MPI application that you run frequently. What is the typical number of MPI processes per job that you run?
(Select all that apply)

1-16 MPI processes
17-64 MPI processes
65-512 MPI processes
513-2048 MPI processes
2049 MPI processes or more
I don't know

Apps

Using the same MPI application from the previous question, what is the typical number of MPI processes that you run per node?
(Select all that apply)

1 MPI process
2-3 MPI processes
4-7 MPI processes
8-15 MPI processes
16 MPI processes or more
I don't know

Apps

Using the same application from the previous question, is it a 32 or 64 bit application?
(Select all that apply)

32 bit
64 bit
I don't know
Other

Other (text)

Apps

I expect to be able to upgrade to an MPI-3 implementation and still be able to run my legacy MPI applications *without recompiling*.

Strongly Disagree
Disagree
Undecided
Agree
Strongly Agree

Additional comments (text)

Apps

I expect to be able to upgrade to an MPI-3 implementation and only need to recompile my legacy MPI applications *with no source code changes*.

Strongly Disagree
Disagree
Undecided
Agree
Strongly Agree

Additional comments (text)

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE

Text Opinion

What ONE THING would you like to see added or improved in the MPI standard?

MPI Functionality

How much are each of the following sets of MPI functionality used in your MPI applications?

Point to point
Collective
Derived / complex datatypes
Comms. other that MCW
Grah or Cartesian comms.
Err. handles other than FATAL

Dynamic MPI procs.
One-sided
MPI_THREAD_MULTIPLE (multiple thr. simul. using MPI)
Multiple thrs., but only 1 in MPI at a time

1	2	3	4	5
Not used at all	Trivially used in some places	Used moderately w/ other MPI functionality	Used heavily w/ other MPI functionality	Comprises the backbone of my application

In the above question, if you marked any set with "Not used at all" or "Trivially used," please explain why. (text)

MPI Functionality

Which of the following do any of your MPI applications use?
(Select all that apply)

Threads
OpenMP
Shmem
Global arrays
Co-processors / accelerators
PGAS languages
I don't now
Other

Other (text)

MPI Functionality

When answering the following question, please remember that that C++ MPI applications can use the C++ and/or C MPI bindings.

Do you have any MPI applications that are both written in C++ and use the MPI C++ bindings?

Yes
No
I don't know

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE

MPI Functionality

The following question refers to the ability to use extremely large count values with MPI operations such as sending/receiving, file actions, and one-sided operations. It makes the assumption that the largest value that a signed C "int" and a default Fortran INTEGER can represent is 2 billion.

My MPI application would benefit from being able to reference more than 2 billion items of data in a single MPI function invocation.

1	2	3	4	5
Strongly disagree	Disagree	Undecided	Agree	Strongly agree
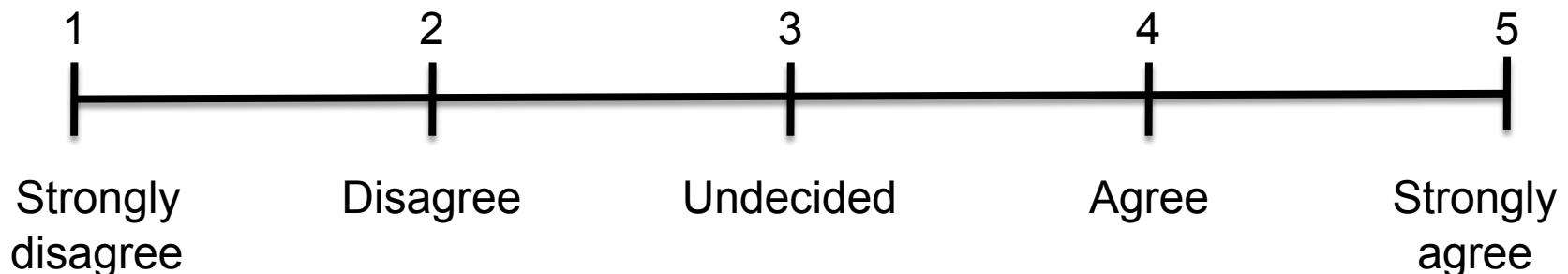
If you answered "Agree" or "Strongly Agree" to the above question, please explain why. (text)

MPI Functionality

One-sided remote memory access (RMA) is an advanced MPI concept. The following question assumes familiarity with the complex issues involved and deliberately makes you choose between two options that may or may not be mutually exclusive. The goal is to find out which is more important to you, regardless of whether they are mutually exclusive or not. If you are unsure how to answer and/or are unfamiliar with MPI RMA concepts, feel free to leave this question unanswered.

MPI one-sided communication performance (e.g., message rate and latency) is more important to me than supporting a rich remote memory access (RMA) feature set (e.g., communicators, datatypes).

1	2	3	4	5
Strongly disagree	Disagree	Undecided	Agree	Strongly agree
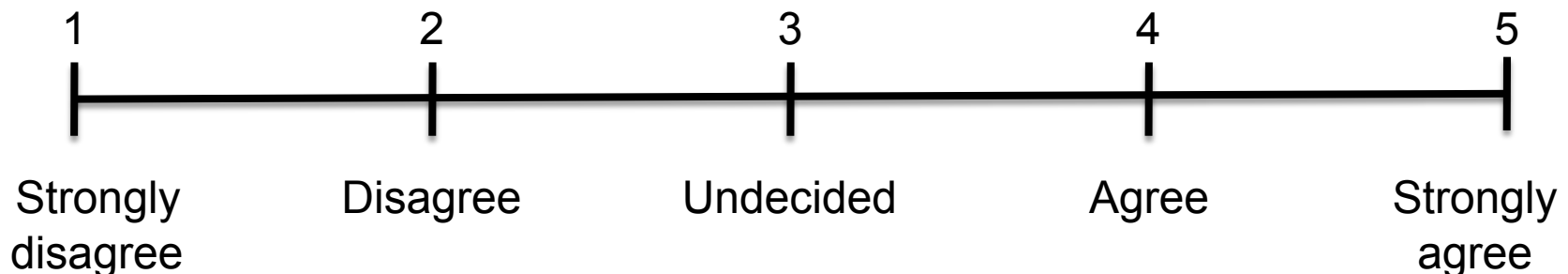
Please explain your rating to the above question: (text)

MPI Functionality

The MPI standard provides certain semantic guarantees that may not be required by a particular application. It also provides functions that many applications never use. The MPI Forum is considering an "assertions" interface that would let an application identify specific functionality it does not depend on, such that an MPI library could improve performance or reduce memory usage by disabling that specific functionality.

The described "assertions" interface would be valuable to my MPI applications.

Question is leading
Discarded

1	2	3	4	5
Strongly disagree	Disagree	Undecided	Agree	Strongly agree
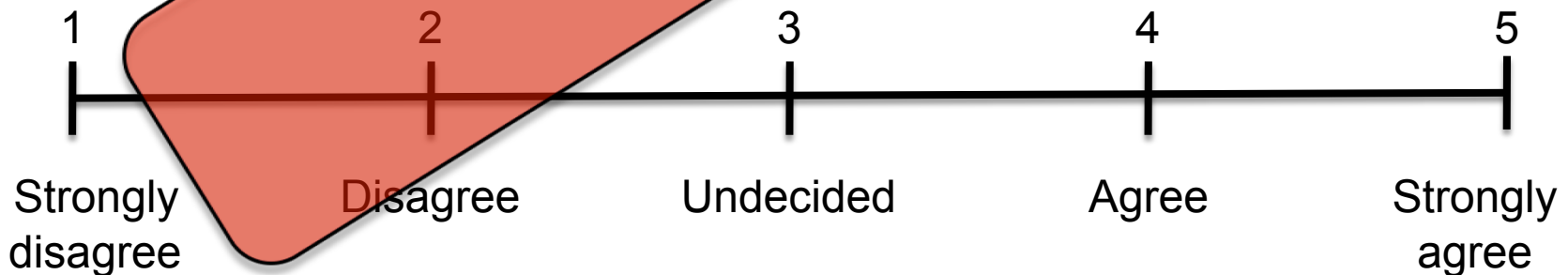
MPI 3 Possibilities

The following is a broad list of topics that the MPI Forum is considering for MPI-3. Note that it is probably safe to assume that using any of the new functionality will involve at least some degree of change to your existing MPI application (e.g., it is unlikely that MPI-3 applications will automatically become fault tolerant; it is much more likely that you will need to add additional fault tolerant logic using new MPI-3 API functions). If you are unfamiliar with a given topic, feel free to leave its rating blank.

MPI 3 Possibilities

Rank the following in order of importance to your MPI applications (1=most important, 6=least important):

Non-blocking collectives
Revamped one-sided (compared to (MPI-2.2)
MPI app. control of fault tolerance
New Fortran bindings
"Hybrid" programming (MPI in conjunction with threads, OpenMP, etc.)
Standardized 3rd party MPI tool support

MPI Functionality

Rate the following in order of importance to your MPI applications (1=most important, 5=least important):

Runtime performance (e.g., latency, bandwidth, resource consumption, etc.)
Feature-rich API
Run-time reliability
Scalability to large number of MPI processes
Integration with other middleware, communication protocols, etc.

Text Opinion

Use the space below to provide any other information, suggestions, or comments to the MPI Forum.

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE

Obvious Survey Results

Presented at the MPI Forum in Atlanta

Obvious Results from Atlanta Meeting

I expect to be able to upgrade to an MPI-3 implementation and still be able to run my legacy MPI applications *without recompiling*.

Strongly Disagree	**257**	
Disagree	**372**	83%
Undecided	**198**	
Agree	114	
Strongly Agree	59	

Obvious Results from Atlanta Meeting

I expect to be able to upgrade to an MPI-3 implementation and only need to recompile my legacy MPI applications *with no source code changes*.

Strongly Disagree	31
Disagree	76
Undecided	**154**
Agree	**394**
Strongly Agree	**341**

89%

Unanswerable Questions from Data

From questions posed by the MPI Forum in Atlanta.

Unanswerable Questions from Data

▸ Of respondents running with >8 ppn, are they counting threads as processes?

  ▸ We cannot make assumptions about respondent interpretation of this question

▸ Why did people mark OpenMP but not Threads?

  ▸ Or: why didn't they mark both OpenMP and Threads?

  ▸ The data does not specify respondent intent

  ▸ We <u>speculated</u> that they viewed OpenMP as a different, higher-level abstraction than threads

Unanswerable Questions from Data

- How do the Fortran users feel about backwards compatibility?
  - Fortran developers were not identified by the data
  - Respondents were asked to value new Fortran bindings – but that says nothing about whether they were Fortran developers
- Of respondents that do not program in Fortran, how do they rank 'new Fortran' bindings? Of those respondents how many are C++ programmers?
  - Neither Fortran nor C++ developers were identified by the data

Unanswerable Questions from Data

▸ How did respondents answer the *recompile only* question in comparison with 'count' question?

  ▸ We analyzed this at length and concluded that the correlation is not meaningful

▸ Rationale:

  ▸ The "recompile only" question is specifically about legacy applications

  ▸ The "MPI count / 2 billion" question is specifically about new MPI-3 functionality

  ▸ In short: straight recompiling (or not) of legacy applications has nothing to do with desire or intent to adopt new MPI-3 functionality

Deeper Analysis

Questions posed by the MPI Forum in Atlanta.

Some General Notes

▶ **`count`**

  ▶ Total of the row or column

▶ **`%R`**

  ▶ % of subset of the respondent population **considered for this question**

▶ **`%T`**

  ▶ % of **all** survey respondents

▶ **`blank`**

  ▶ Respondents that did not answer this question

▶ **`null`**

  ▶ The respondent dropped out before reaching this question

Data Slicing: Demographics

▸ Which of the following best describes you?

▸ Rate your expertise with the MPI standard.

▸ 1 (Not Familiar), 2 (basic), 3 (good), 4 (deep), 5 (expert)

```
             | 1  |  2  |  3  | 4  | 5 | blank| Row Total| % Total
-----------+----+-----+-----+----+---+-----+---------+-------
  MPI User | 9  | 92  |  44 | 11 | 2 |   1 |    159  | 11.35
  App Devel| 3  | 56  | 193 | 49 | 2 |   0 |    303  | 21.63
Middleware | 1  | 12  |  51 | 35 | 5 |   0 |    104  |  7.42
Implementer| 0  | 13  |  20 | 20 | 1 |   0 |     54  |  3.85
  Academic | 12 | 106 | 127 | 44 | 4 |   2 |    295  | 21.06
   Student | 14 | 47  |  36 |  5 | 0 |   1 |    103  |  7.35
Management | 1  | 10  |  13 |  4 | 3 |   0 |     31  |  2.21
     Other | 2  |  9  |   8 |  6 | 0 |   0 |     25  |  1.78
     blank | 0  |  2  |   0 |  0 | 0 | 325 |    327  | 23.34
         % |3.0 | 24.8| 35.1| 12 | 1 | 23.5|
```

What type of user runs with >8ppn?

▸ 1 (Not Familiar), 2 (basic), 3 (good), 4 (deep), 5 (expert)

```
            | 1  |  2  |  3  | 4  | 5 | blank| R.Total| %R | %T
------------+----+-----+-----+----+---+------+--------+----+----
  MPI User  | 2  | 39  | 25  | 7  | 1 |  1   |   75   | 19 |  5   ⬅
 App Devel  | 1  | 21  | 79  | 28 | 1 |  0   |  130   | 34 |  9   ⬅
Middleware  | 0  |  3  | 13  | 13 | 2 |  0   |   31   |  8 |  2
Implementer |  0 |  6  | 12  | 6  | 1 |  0   |   25   |  6 |  2
 Academic   | 3  | 27  | 36  | 14 | 0 |  1   |   81   | 21 |  6   ⬅
  Student   | 0  |  8  |  8  | 2  | 0 |  0   |   18   |  5 |  1
Management  | 0  |  3  |  8  | 2  | 2 |  0   |   15   |  4 |  1
    Other   | 0  |  5  |  4  | 3  | 0 |  0   |   12   |  4 |  1
    blank   | 0  |  1  |  0  | 0  | 0 |  0   |    1   |  0 |  0
```

▸ Peak at Application Developers, followed by Academic and User.

▸ Our interpretation: Leading the way for more cores?

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE

What Respondents Said OpenMP But Not Threads?

▸ 1 (Not Familiar), 2 (basic), 3 (good), 4 (deep), 5 (expert)

```
             | 1  |  2  |  3  | 4  | 5 | blank| R.Total| %R | %T
------------+----+-----+-----+----+---+------+--------+----+----
   MPI User | 1  | 19  | 12  | 0  | 1 |  0   |   33   | 15 |  2
  App Devel | 0  | 11  | 56  | 24 | 1 |  0   |   92   | 41 |  7
 Middleware | 0  |  1  |  7  | 3  | 1 |  0   |   12   |  5 |  1
Implementer | 0  |  1  |  5  | 2  | 0 |  0   |    8   |  4 |  1
   Academic | 1  | 19  | 26  | 6  | 1 |  0   |   53   | 23 |  4
    Student | 0  |  6  | 10  | 0  | 0 |  0   |   16   |  7 |  1
 Management | 0  |  1  |  3  | 0  | 0 |  0   |    4   |  2 |  1
      Other | 0  |  2  |  3  | 2  | 0 |  0   |    7   |  3 |  1
      blank | 0  |  1  |  0  | 0  | 0 |  0   |    1   |  0 |  0
```

```
           | count
-----------+-------
   blank    |   223
   CUDA     |     1
   TBB      |     1
   mpich2   |     1
```

▸ Of those that marked 'Other', what did they write?

What Respondents Said OpenMP and Threads?

▸ 1 (Not Familiar), 2 (basic), 3 (good), 4 (deep), 5 (expert)

```
            | 1 |  2 |  3 | 4  | 5 | blank| R.Total | %R | %T
------------+---+----+----+----+---+------+---------+----+----
  MPI User  | 0 | 14 |  6 |  4 | 1 |   0  |    25   | 11 |  2
  App Devel | 0 | 10 | 34 | 15 | 1 |   0  |    60   | 27 |  4
  Middleware| 0 |  2 | 13 | 14 | 2 |   0  |    31   | 14 |  2
Implementer | 0 |  2 |  4 |  8 | 0 |   0  |    14   |  6 |  1
  Academic  | 0 | 22 | 20 | 20 | 2 |   0  |    64   | 28 |  5
   Student  | 0 |  7 |  7 |  3 | 0 |   0  |    17   |  8 |  1
 Management | 0 |  1 |  3 |  2 | 3 |   0  |     9   |  4 |  1
     Other  | 0 |  2 |  3 |  0 | 0 |   0  |     5   |  2 |  0
```

Respondents who said "yes" to OpenMP and / or Threads

▸ What size nodes do they use?

  ▸ Seems to match current popular hardware trends

```
   1   | 2-3 | 4-7 | 8-15 | >=16 | I don't know
 -----+-----+-----+------+------+-------------
  189 | 192 | 265 | 190  | 78   | 12
```

Respondents who said "yes" to OpenMP and/or Threads

MPI_THREAD_MULTIPLE
(multiple threads simultaneously
using MPI)

	count	%R	%T
Not at all	260	52	19
Trivially	65	13	5
Moderately	81	16	6
Heavily	60	12	4
Backbone	33	7	2

Multiple threads, but only one in
MPI at a time.

	count	%R	%T
Not at all	179	36	13
Trivially	83	17	6
Moderately	121	24	6
Heavily	78	16	6
Backbone	34	7	2

▸ How many respondents marked 'Not at all' for both of these choices, but still 'OpenMP' or 'Threads'?

'confused'	%R
141	10

Respondents who said "yes" to OpenMP and/or Threads (Exclude 'confused')

MPI_THREAD_MULTIPLE (multiple threads simultaneously using MPI)

	count	%R	%T
Not at all	260	52	19
Trivially	65	13	5
Moderately	81	16	6
Heavily	60	12	4
Backbone	33	7	2

	count	%R	%T
Not at all	119	33	8
Trivially	65	18	5
Moderately	81	23	6
Heavily	60	17	4
Backbone	33	9	2

Multiple threads, but only one in MPI at a time.

	count	%R	%T
Not at all	179	36	13
Trivially	83	17	6
Moderately	121	24	6
Heavily	78	16	6
Backbone	34	7	2

	count	%R	%T
Not at all	38	11	3
Trivially	83	23	6
Moderately	121	34	9
Heavily	78	22	6
Backbone	34	10	2

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE

OpenMP/Threads with >8ppn

▸ Users running >8ppn
  ▸ 388

▸ Of those, how many chose OpenMP and/or Threads
  ▸ 230 (Including confused)
  ▸ 175 (Excluding confused)

▸ Of the 388, how many chose neither OpenMP nor Threads
  ▸ 158

Do users of larger systems care more about Parallel I/O?

	1-16	17-64	65-512	513-2048	>2048	donno	count	%T
Not at all	148	158	142	67	44	5	314	22
Trivially	37	54	59	28	16	0	107	8
Moderately	73	73	89	43	35	2	180	13
Heavily	54	60	60	40	35	1	129	9
Backbone	16	15	16	12	14	1	36	3
blank	144	135	100	34	30	2 9	635	45

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE

Who uses C++ bindings?

▸ 1 (Not Familiar), 2 (basic), 3 (good), 4 (deep), 5 (expert)

```
Population: 165 of 1401
            | 1 | 2  | 3  | 4  | 5 | null | count | %R | %T
-----------+---+----+----+----+---+------+-------+----+-----
  MPI User | 0 | 15 |  7 |  1 | 2 |    0 |    25 | 15 |   2
 App Devel | 1 | 12 | 21 |  9 | 0 |    0 |    43 | 26 |   3
Middleware | 0 |  2 |  5 |  6 | 0 |    0 |    13 |  8 |   1
Implementer| 0 |  1 |  2 |  6 | 0 |    0 |     9 |  5 |   1
  Academic | 0 | 15 | 24 | 11 | 2 |    0 |    52 | 32 |   4
   Student | 0 |  6 |  9 |  1 | 0 |    0 |    16 | 10 |   1
Management | 0 |  1 |  1 |  0 | 2 |    0 |     4 |  2 |   0
     Other | 0 |  1 |  0 |  2 | 0 |    0 |     3 |  2 |   0
```

▸ No obvious grouping

▸ Small – but nonzero populations in several areas

Compare 'one sided' importance to Performance vs. Functionality RMA Q.

1 = Strongly Disagree (Functionality)
2 = Disagree
3 = Undecided
4 = Agree
5 = Strongly Agree   (Performance)

one-sided

	1	2	3	4	5	*blank*	count	%T
None	8	28	112	54	27	147	376	27
Trivially	2	9	50	32	16	45	154	11
Moderately	1	16	32	42	16	51	158	11
Heavily	0	3	6	16	8	6	39	3
Backbone	1	2	5	7	2	2	19	1
blank	1	1	40	9	2	602	655	47

▸ Peaks at columns 3 and 4

▸ Of heavy users, only slight lean to performance

Same for only ARMCI/Global Arrays

```
1 = Strongly Disagree (Functionality)
2 = Disagree
3 = Undecided
4 = Agree
5 = Strongly Agree    (Performance  )
```

one-sided	1	2	3	4	5	*blank*	count	%R	%T
Not at all	2	5	9	3	6	11	36	34	3
Trivially	0	2	9	4	2	7	24	22	2
Moderately	1	1	5	5	2	8	22	21	2
Heavily	0	1	1	5	4	0	11	10	1
Backbone	1	1	1	2	0	0	5	5	0
blank	0	0	2	2	0	5	9	8	1

▸ Data requested by ARMCI Community

Compare 'Run-time Performance' to Nonblocking Collectives (NBC)

```
1    = Most Important
2-5 = ...
6    = Least Important        Nonblocking Collective (NBC)
```

	blank	1	2	3	4	5	6	null	total	%T
blank	96	5	1	2	1	0	0	0	105	7
1 (Most)	40	152	68	50	40	30	17	0	397	28
2	27	58	48	36	23	9	5	0	206	14
3	13	21	11	24	15	4	1	0	89	6
4	2	6	6	6	3	1	3	0	27	2
5 (Least)	3	1	1	2	4	1	2	0	14	1
	0	0	0	0	0	0	0	563	563	40
Total	181	243	135	120	86	45	28	563		
%T	13	17	10	9	6	3	2	40		

(Row axis label: Run-time Performance)

▸ Those that value NBC also value performance

Compare 'Feature-rich API' to NBC

```
1   = Most Important
2-5 = ...
6   = Least Important       Nonblocking Collective (NBC)
```

Feature-Rich API

	blank	1	2	3	4	5	6	null	total	%T
blank	123	19	8	9	1	1	1	0	162	12
1 (Most)	3	1	3	4	0	1	2	0	14	1
2	1	16	7	8	4	0	2	0	38	3
3	1	26	16	14	8	2	3	0	70	5
4	28	98	46	47	38	18	8	0	283	20
5 (Least)	25	83	55	38	35	23	12	0	271	19
	0	0	0	0	0	0	0	563	563	40
Total	181	243	135	120	86	45	28	563		
%T	13	17	10	9	6	3	2	40		

▸ Our interpretation: positive response for NBC, respondents seem to view it as a performance API not just a new feature API

Compare NBC to {Perf., Feature-rich API}

▸ Those who want NBC:

  ▸ Value run-time performance highly

  ▸ Do not value a feature-rich API as highly

▸ Our Interpretation:

  ▸ View NBC as a performance enhancement instead of just a new API

    ▸ Users will assume that NBC will perform well

    ▸ Users will assume that NBC implementation will provide overlap

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE

Questions for the MPI Forum

Distribution of interpretation and data

▸ These slides will be made available publicly

  ▸ It is unlikely that we will turn this into a written report

▸ Additional analysis is ongoing

  ▸ E.g., we have not yet analyzed the text answers (there's a LOT!)

▸ Raw data from the survey will be publicly available

  ▸ We want (and encourage) others to analyze it

  ▸ Numeric data will be available shortly

  ▸ Textual data must be "cleaned" of personally identifiable information first (this will take some time)

▸ Look for link to the data on the MPI Forum wiki site

  ▸ https://svn.mpi-forum.org/trac/mpi-forum-web/wiki

INDIANA UNIVERSITY
PERVASIVE TECHNOLOGY INSTITUTE