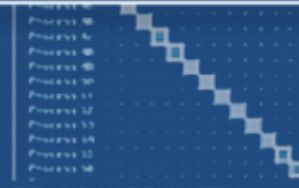




SOFTWARE

+ 19.56 updatex
+ 399.70 updateien
+ 0.00 gene
- 0.00 <<iteration loop>>
+ 447.52 genbc

PRODUCTIVITY



FAST SOLUTIONS

- PAPI_L1_ICM
- PAPI_L2_DCM
- PAPI_L2_ICM
- PAPI_L1_TCM

Marmot MPI Correctness Tool: Details and Further Examples

For the 4th VI-HPS Tuning Workshop

Tobias Hilbrich
GWT-TUD GmbH

tobias.hilbrich@zih.tu-dresden.de

September 2009

- Further Examples
- Details on Building with Marmot
- Environmental Setup

- Available in C and Fortran
- Use 2 processes (3 with Marmot)
- For C:
 - Source code information not available, details on that are given on later slides
- Compile & run with Marmot and interpret Marmot's output
- *Error & Solution see next slide*

- Error:
 - Usage of a non-committed datatype in MPI_Send and MPI_Recv
- Solution:
 - Add a “MPI_Type_commit” (e.g.: after the MPI_Type_contiguous)

- Available in C and Fortran
- Use 2 processes (3 with Marmot)
- For C:
 - Source code information not available, details on that are given on later slides
- Compile & run with Marmot and interpret Marmot's output
- *Error & Solution see next slide*

- Error:
 - Processes 0 and 1 deadlock
- Solution:
 - For one of the processes change the order of the send and the recv

- Available in C
- Use two runs one with 4, one with 5 processes (5 or 6 with Marmot)
- More complex example
 - Performs a 2D border exchange using a Cartesian communicator
- For C:
 - Source code information not available, details on that are given on later slides
- Compile & run with Marmot and interpret Marmot's output
- *Error & Solution see next slide*

- Errors:
 - 1) Works only for process counts that are perfect squares (e.g. 4)
 - 2) Uses Fortran MPI datatype in C
 - 3) Missing MPI_Type_commit
 - 4) Message buffers (Isend/Irecv) overlap
 - 5) Two datatypes and one communicator not freed
- Solution:
 - 1) Use MPI_Dims_create, or do not use processes that get an MPI_COMM_NULL returned from MPI_Cart_create
 - 2) Replace MPI_INTEGER with MPI_INT
 - 3) Add MPI_Type_commit twice (e.g.: at end of initializeTypes)
 - 4) Use Bsend or Sendrecv (not so easy)
 - 5) Add a MPI_Comm_free and two MPI_Type_free calls

- C and C++:
 - Use `marmotcc` or `marmotcxx`
 - Tries to automatically add source code locations to MPI calls
 - This fails for some MPI implementations, e.g., OpenMPI
 - Workaround modify all places where `mpi.h` is included:
 - ```
#include <mpi.h>
#ifdef MARMOT
#include <enhancempicalls.h>
#include <sourceinfompicalls.h>
#endif
```
    - Compile with “`marmotcc -DMARMOT`”
  - To see what the compiler wrapper does use “`--marmot-verbose`”

- Fortran (1/2):
  - Use [marmotf77](#) or [marmotf90](#)
  - Adds source locations by instrumenting your source code
  - Help:

```
"marmotf77" Help:
usage: marmotf77 [--help] [--marmot-verbose] [--marmot-fixed]
 [--marmot-free] [--marmot-noinst]
 [--marmot-keep-instrumented-files] <arguments>
Replace your Fortran77 compiler with this script.
--marmot-verbose : prints extra information
--help : prints this help
--version : prints the Marmot version
--marmot-fixed : forces instrumentation to analyze source as fixed source fortran
--marmot-free : forces instrumentation to analyze source as free source fortran
--marmot-noinst : disables instrumentations
--marmot-keep-inst-files : forces marmot to not remove temporary instrumented
 files ([NAME].f => __[NAME].f)
...
```

- Use “[--marmot-verbose](#)” to see details on what the tool does

- Fortran (2/2):
  - Compiler wrappers need to guess the Fortran language
  - May guess wrong if you use a uncommon file suffix
    - Use “`--marmot-fixed`” to force analysis as fixed source Fortran
    - Use “`--marmot-free`” to force analysis as free source Fortran
  - You can disable the instrumentation if it fails
    - Use “`--marmot-noinst`”

- Change the default behaviour of Marmot with env. variables
- First part of Marmot logfiles specifies the used values
- Important environment variables are:
  - MARMOT\_LOGFILE\_TYPE
    - Sets output type (0=Text, 1=HTML, 2=CUBE)
  - MARMOT\_LOG\_FILTER\_COUNT
    - Limits how often a certain Marmot output may is repeated
  - MARMOT\_LOG\_FLUSH\_TYPE
    - Causes Marmot to flush output files immediately  
(Useful if application crashes but no Marmot output present)
  - MARMOT\_LOGFILE\_NAME
    - Name of the output file
  - MARMOT\_LOGFILE\_PATH
    - Path for the output file

- Marmot is OpenSource, available at:

<http://www.hlrs.de/organization/av/amt/research/marmot/>

- For questions or support use:

[marmot-supp@gforge.hlrs.de](mailto:marmot-supp@gforge.hlrs.de)