

## ***Interactive comment on “Influence of the compiler on multi-CPU performance of WRFv3” by T. Langkamp***

**T. Langkamp**

thomas.langkamp@uni-hamburg.de

Received and published: 18 May 2011

Dear Mr. Morton,

first of all thank you for your constructive and helpful comments. Now my answer in detail:

You wrote: "Page 548, Line 5: Though "This is currently the most used hardware..." is probably correct, a citation would be nice. Page 551, Line 17: Though the claim that ARW is recommended for research and NMM for operational, I think this needs at least a citation and maybe a little qualification. Page 552, Line 18: The claim that storage and login nodes are "non-crucial components in respect of performance" (should be "...with respect to performance...") might be valid for this particular study, is definitely

C259

not true in more general cases. Maybe this should be reworded for clarity."

I agree to all three comments and will include the needed citations and clarifications.

You wrote: "Page 552, Line 23: The statement is made that "WRF is definitely influenced by the old kernel version." I suggest that this statement "teases" the reader, who may want to understand why (and how) this is the case. I recommend it either be deleted or explained more fully."

The kernel and its' included hardware-drivers are influencing the system-performance in many ways. To get a good overview one should look at the numerous benchmarks on phoronix.com. They tested the linux-kernels of the last 5 years (which includes the used 2.6.16 kernel) with a broad benchmark-suite, including benchmarks for Disk- and Network-I/O, compression, encoding, encryption, rendering etc. To pick out one relevant benchmark, one should look at the results of the "IS.C test" within NASA's "NAS Parallel Benchmarks" (NPB) ([http://www.phoronix.com/scan.php?page=article&item=linux\\_2612\\_2637&num=3](http://www.phoronix.com/scan.php?page=article&item=linux_2612_2637&num=3)).

This is a small set of programs especially designed to evaluate the performance of parallel supercomputers. The old kernel shows about 4% performance degradation over a current (2.6.3x) kernel version. Phoronix also discusses the numerous performance-related changes throughout the kernel versions. I will include this information as a footnote or only the link in the revised paper.

You wrote: "Page 553, Line 18: The author makes interesting - apparently verified - statements about the effects of -O0 and -O3 optimizations on C++ and Fortran code. In our past experiences, -O3 optimizations have sometimes introduced instabilities in the code that seem to get resolved when going back to -O0 optimizations. This has been encountered by us on a number of occasions, and I believe it might warrant mentioning in here that these optimizations just might end up being a source of hard-to-detect problems."

Even if another reviewer already mentioned that there are "unsafe" compiler-options in

C260

general, it is surprising news for me that this is the case for the "-O3" optimization, as it is enabled by default for Intel compilers. I did not know about the phenomenon and would like to read more about it to include this information in the paper; would you mind recommending some papers on this topic to read?

You wrote: "Page 554, Line 20: Somewhere in this area, it might be worth suggesting that large memory support be a "default" setting for netCDF and WRF when compiling. More than once, we have watched people struggle with seemingly mysterious I/O issues that were finally resolved when it was deduced that the output files were larger than 2 GBytes. Although not a factor in this work, it's one of those "hidden" pieces of info that might be prudent to include in searchable articles."

This information will be included.

You wrote: "Page 556, Line 13: The author discusses a a mysterious problem in slow-down when using Tornado on 64 cores. Our group has come across two different Sun clusters (and, Tornado appears to be a Sun cluster) in the past year in which this kind of behaviour was tracked down to bad nodes. Isolating the bad node(s) resolved the problem (though users need to be wary that any of the nodes can go bad at any time)."

As I repeated the benchmark many times on different nodes, this seems not to be the case here.

Yours sincerely, Thomas Langkamp

---

Interactive comment on Geosci. Model Dev. Discuss., 4, 547, 2011.