

Met Office WAVEWATCH III profiling and scale-ability activities

Andy Saulter Presentation to WWIII developers group 5th June 2019



Why are the Met Office doing this?

- Component of wider 'next generation modelling systems' project assessing model lifecycle with respect to future computing architecture and science requirements:
 - Unified Model LFRic and JEDI
 - NEMO and NEMOVar
 - WAVEWATCH III
- For WAVEWATCH, focus is on two aspects:
 - Optimization and scale-ability on existing architecture (next HPC)
 - Potential to run on next generation HPC architecture (e.g. GPUs)

The scale-ability problem: MPI

- WAVEWATCH switches between two modes of MPI decomposition:
 - Source term steps decomposed by spatial grid location
 - Propagation steps decomposed by spectral bin (for regular grids, the unstructured grid has an extra level of domain decomposition using PDLIB)
- The propagation decomposition theoretically limits the number of MPI ranks that the model can use to the number of spectral bins
- In fact load balancing requirements reduces this by ~factor of 3; push it too hard and you get this message!

```
!/DIST 8029 FORMAT (/' *** WAVEWATCH III ERROR IN W3INIT : '/ &
!/DIST ' SOMETHING WRONG WITH MPP PROPAGATION MAP.'/ &
!/DIST ' CALL HENDRIK !!!'/)
```

 This cap on the number of processors we can use limits standalone model size in operational systems with strict time slots and has the potential to make WAVEWATCH the weak link in coupled systems where other models are more scale-able

The scale-ability problem: MPI-OpenMP

- One option for increasing processor limits without resorting to propagation spatial domain decomposition is to use hybrid MPI-OpenMP calls; i.e. processors = number of procs per rank x number MPI ranks
- Tests with Met Office global wave model (25-12-6-3km SMC grid using a 30x36 freq-dirn bin spectrum) indicate that some efficiencies can be achieved but there is still a cap...



Instrumenting WAVEWATCH III

- In order to better analyse how far we can push MPI-OpenMP (and/or find other options for model optimization) Met Office are undertaking a process of instrumenting WAVEWATCH III with 'Dr Hook' (no not that one!)
- Dr Hook is a lightweight method for traceback and profiling models; principally developed at ECMWF (<u>https://software.ecmwf.int/wiki/download/attachments/19661682/drhook.pdf</u>) and now used as a key tool in profiling the Met Office Unified Model
- Revisions to the code comprise addition of Dr Hook's profiling modules and adding calls around the routines you wish to profile; e.g.

```
w3psmcmd.ftn:!/HOOK CHARACTER(LEN=*), PARAMETER :: routinename = 'SMCDCXY'
w3psmcmd.ftn:!/HOOK INTEGER(KIND=jpim), PARAMETER :: zhook_in = 0
w3psmcmd.ftn:!/HOOK INTEGER(KIND=jpim), PARAMETER :: zhook_out = 1
w3psmcmd.ftn:!/HOOK REAL(KIND=jprb) :: zhook_handle
w3psmcmd.ftn:!/HOOK CALL dr_hook(RoutineName,zhook_in,zhook_handle)
.....WWIII SUBROUTINE CODE
```

w3psmcmd.ftn:!/HOOK CALL dr_hook(RoutineName,zhook_out,zhook_handle)

Results for SMC grid model

• Sample of Dr Hook output

Profiling information for program='./ww3_shel', proc#1:

No. of instrumented routines called : 48

Instrumentation overhead: 2.37%

Memory usage : 1244 MBytes (heap), 969 MBytes (rss), 0 MBytes (stack), 0 (paging)

Wall-time is 1312.34 sec on proc#1 (36 procs, 3 threads)

- Thread#1: 1312.15 sec (99.99%)
- Thread#2: 361.26 sec (27.53%)
- Thread#3: 359.05 sec (27.36%)
- # % Time Cumul Self Total # of calls Self Total Routine@<thread-id>

(Size; Size/sec; Size/call; MinSize; MaxSize)

	(self)	(sec) (sec) ((sec)	ms/call	ms/call			
1	41.51	544.813	544.813	544.817	2880	189.17	189.17	W3IOPE@1	
2	19.28	797.829	253.016	253.164	491600	0.51	0.51	*W3SRCE@3	
3	19.28	797.829	252.998	253.151	488694	0.52	0.52	W3SRCE@2	
4	19.28	797.829	252.984	253.133	493621	0.51	0.51	W3SRCE@1	
5	16.54	1014.870	217.04	1 217.044	2880	75.36	75.36	W3GATHSMC_rendezvous	s_2@1

Results for SMC grid model

WW3 DrHook Routine Walltime Percentage

• Profile changes: Load imbalance at high processor counts

■ W3SRCE@1

20 Node, 300 MPI Ranks, 1 OpenMP thread per rank

SMCxUNO2@1 SMCyUNO2@1 W3OUTG@1 SMCAverg@1 INIT GET ISEA@1 W3GATHSMC rendezvous 2@1 W3KRTN@1 W3WAVE@1 W3WAVE rendezvous 3@1 Sdrhook@1 W3IORS rendezvous 2@1 W3UWND@1 W3IOGR@1 W3FLDG@1 ■ W3SCATSMC@1 W3SCATSMC rendezvous 2@1 W3WAVE rendezvous 2@1 W3GATHSMC@1 SMCDHXY@1 W3SHEL rendezvous 5@1 W3INIT@1 W3SHEL_rendezvous_4@1 W3FLDO@1 ■ W3SHEL rendezvous 2@1 W3IOTR@1 W3WAVE_rendezvous_1@1 W3UICE@1 W3SHEL@1 ■ INIT GET JSEA ISPROC@1 W3IORS@1 WWTIME@1 W3SHEL rendezvous 1@1 W3IOTR_rendezvous@1 STME21@1 ■ W3ACTURN@1 NEXTLN@1 W3WAVE rendezvous 6@1 W3READFLGRD@1 W3SCATSMC_rendezvous_1@1 = TICK21@1 ■ W3SETG@1 W3SETA@1 ■ W3SET@1 W3NDAT@1 ■ W3NAUX@1 W3NINP@1 W3WAVE rendezvous 9@1 W3SETW@1 W3NMIN@1 W3WAVE_rendezvous_8@1

 WWBSKCg1

 SMGUNO2013

 WWGATHSMC_memorycous_2011

 WWGATHSMC_memorycous_2011

 WWGATHSMC_memorycous_2011

 WWGATHSMC_memorycous_2011

 WWGATHSMCC01

 WWGATHSMC020

 WWGATHSMC020

WW3 DrHook Routine Walltime Percentage 20 Node, 720 MPI Ranks, 1 OpenMP thread per rank

> ■ W3WAVE rendezvous 2@1 ■ W3SRCE@1 W3PSMC@1 W3WAVE@1 SMCyLINO2@1 SMCyUNO2@1 ■ W3FLDG@1 W3GATHSMC_rendezvous_2@1 ■ INIT GET ISEA@1 W3OUTG@1 W3IORS rendezvous 2@1 W3WAVE_rendezvous_3@1 SMCAvera@1 W3KRTN@1 Sdrhook @1 W3UWND@1 ■ W3IOGR@1 W3SCATSMC_rendezvous_2@1 SMCDHXY@1 W3SCATSMC@1 ■ W3INIT@1 W3SHEL rendezvous 5@1 ■ W3SHEL rendezvous 4@1 ■ W3GATHSMC@1 ■ W3SHEL @1 ■ W3SHEL rendezvous 1@3 W3FLDO@1 ■ W3WAVE_rendezvous 1@1 W3IOTR rendezvous@1 W3UICE@1 INIT_GET_JSEA_ISPROC@1 = W3IORS@1 ■ WWTIME@1 STME21@1 W3IOTR@1 NEXTLN@1 ■ W3READFLGRD@1 W3ACTURN@: TICK21@1 ■ W3SHEL rendezvous 2@1 ■ W3SFTG@ W3WAVE rendezvous 6@1 W3SETA@1 W3NDAT@1 ■ W3NAUX@1 W3SET/@1 ■ W3SETW@ W3NMIN@1 ■ W3NINP@1 W3WAVE_rendezvous_8@1

360 procs: Source and propagation term dominated

720 procs: Large waiting times (end of propagation step)

Results for SMC grid model

• Profile changes: MPI vs OpenMP

WW3 DrHook Routine Walltime Percentage



■ W3SRCE @1 W3PSMC@1 W3GATHSMC rendezvous 1@1 SMCxUNO2@1 W3OUTG@1 SMCyUNO2@: SMCAvera@1 INIT GET ISEA@1 W3KRTN/01 Sdrbook @1 W3WAVE@1 W3SCATSMC rendezvous 2@1 ■ W3GATHSMC@1 ■ W3SCATSMC@1 W3GATHSMC_rendezvous_2@1 W3IORS_rendezvous_2@1 ■ W3ELDG@1 W3WAVE rendezvous 1/01 ■W3WAVE rendezvous 2@1 W3UWND@1 W3INIT@1 W3WAVE_rendezvous_3@1 ■ W3IOGR@ SMCDHXY@1 W3SHEL_rendezvous_5@1 W3FLDO@1 W3SHEL rendezvous 4@1 W3IOTR@1 ■ W3WAVE rendezvous 6@1 W3SHEL rendezvous 1@1 W3UICE@1 W3SHEL rendezvous 2@1 INIT_GET_JSEA_ISPROC@1 W3SHEL@1 W3IORS@1 WWTIME@1 ■ W3ACTURN@1 STME21@1 NEXTLN@1 W3SCATSMC rendezvous 1@1 W3IOTR_rendezvous@1 TICK21@1 ■ W3READFLGRD@1 W3SETG@1 ■ W3SETA@1 W3NDAT@1 ■ W3WAVE rendezvous 9@1 ■ W3NAUX@1 W3NINP@1 W3WAVE_rendezvous_7@1 W3SET@1 W3NMIN@1 ■ W3SETW@1 W3WAVE_rendezvous_8@1

MPI: Source and propagation term dominated

WW3 DrHook Routine Walltime Percentage 1 Node, 4 MPI Ranks, 9 OpenMP thread per rank

> ■W3PSMC@1 SMCAvera@: W3OUTG@1 W3SRCE@1 ■ INIT GET ISEA@1 ■ W3SCATSMC@: SMCxUN02@1 ■W3WAVE@1 ■ SMCvUNO2@1 Sdrhook@1 ■ W3GATHSMC@1 *W3KRTN@1 W3GATHSMC_rendezvous_1@1 W3WAVE_rendezvous_4@1 W3IORS rendezvous 2@1 W3UWND@1 W3INIT@1 W3IOGR@1 W3WAVE rendezvous 3@1 W3SHEL rendezvous 5@1 W3GATHSMC rendezvous 2@1 W3SCATSMC rendezvous 2@1 SMCDHXY@1 ■ W3ELDG@1 W3ACTURN@1 W3UICE@1 W3WAVE rendezvous 1@1 W3WAVE rendezvous 2@1 ■ INIT GET JSEA ISPROC@1 ■ W3FLDO@1 W3SCATSMC rendezvous 1@1 W3IORS@1 ■ W3SHEL@1 W3SHEL_rendezvous_1@1 ■ W3IOTR@1 NEXTIN@1 ■ W3IOTR rendezvous@1 W3SETA@1 W3WAVE rendezvous 9@1 ■ W3WAVE rendezvous 7@1 W3SETG@1 STME21@1 ■ W3SHEL rendezvous 3@1 W3READELGRD@1 TICK21@1 W3WAVE_rendezvous_6@1 ■ W3SETW@1 W3NAUX@1 W3SHEL_rendezvous_4@1 ■ W3NDAT@1 W3SET@1 ■W3WAVE rendezvous 8@1 W3NMIN@1 W3NINP@1

MPI-OpenMP: Propagation dominated

Results for SMC grid model

• Profile changes: MPI vs OpenMP scaling by subroutine



1 Node MPI vs OpenMP

Next steps

- Further instrumenting, code optimization of SMC/regular grid propagation modules (if feasible) – possibility for inclusion in vn7?
- Review of domain decomposition options for SMC/regular grids
- Testing on different CPU/GPU flavours:
 - Ongoing testing with ARM processors on Met Office 'Isambard' machine
 - GPU Hackathon in September (Met Office and NVIDIA) probably looking at a subset of code (e.g. SMC propagation module); this may be more about portability than speedup!!