Improvement of TOMCAT-GLOMAP File Access with User Defined MPI Datatypes

Mark Richardson, Numerical Algorithms Group

April 2013

HECToR

RESEARCH COUNCILS UK

What is HECToR?

- Cray XE6
  - 90112 cores
  - 32 cores per node (2xAMD Interlagos processor)
  - 32GB RAM
- Available to UK academics under RCUK
  - EPSRC, BBSRC and NERC (++)
- CSE
  - Help desk – web interface (SAFE)
  - HPCx , help desk staff, system administrators
  - NAG, 12 FTE and some 8 DCSE FTE
- DCSE
  - PI request this support through regular calls for proposals

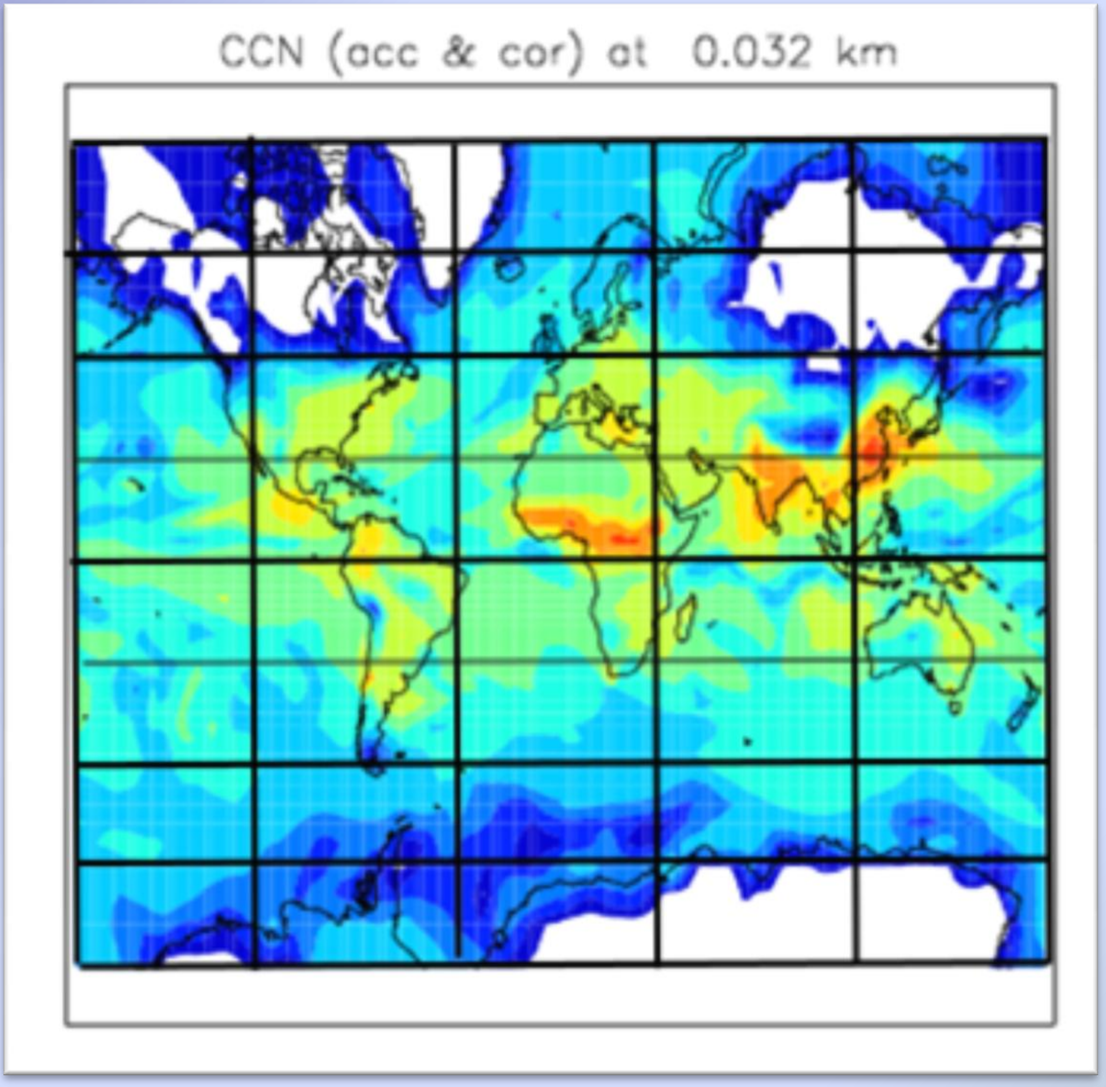
HECToR

RESEARCH COUNCILS UK

Topics covered in this talk

- The analysis of the simulation using the hi-res case
- Focus on two subroutines that have been revised to improve the efficiency of the simulation
  - GBSTAT
  - SORZM
- Two I/O functions have been revised.
  - PPREAD
  - PPWRIT
- Potential future enhancements identified
  - Examination of the NetCDF function

TOMCAT domain decomposition



NPROCI by **NPROCK** patches

Globally 320x160x60 cells
All atmospheric layers contained within MPI task “patch”
i.e. MYLON by MYLAT by NIV grid-boxes

PE80 as supplied is 5x16
where each “patch” is 64x10x60
PE160 is 5x32; each patch is 64x5x60
PE400 is 5x80; each patch is 64x2x60
i.e. Only the number of latitudes is reducing

The NPROCI of 5 fixed by Courant condition near poles of planet: Rotational speed and maximum wind speed require 64 grid-boxes

Analysis: Higher resolution test case

- Code structure
  - Examination of an iteration with no IO shows CONSOM is a significant workload
- Improve the file interaction
  - Earlier DCSE reported that IO appeared inefficient
  - Higher resolution model (T106)
- Examine runtime profile with CrayPAT
  - Actual file access time is low
  - Time is spent around the file accesses
- Review NetCDF
  - How has it been implemented
  - Can it be converted to Parallel
  - What is the alternative?

HECToR

RESEARCH COUNCILS UK

Analysis: Code Structure

- Loop index in hotspots
  - I,K,L,JV indexing SM(I,K,L,JV) as RHS
  - Remove conditionals
- Activate compiler options to tell you what is happening
  - PGI
    - *-Minfo –Mneginfo*
- PAT API to turn on logging for limited sections
  - get fine-grained analysis
  - reduced penalty of huge ap2 files

HECToR

RESEARCH COUNCILS UK

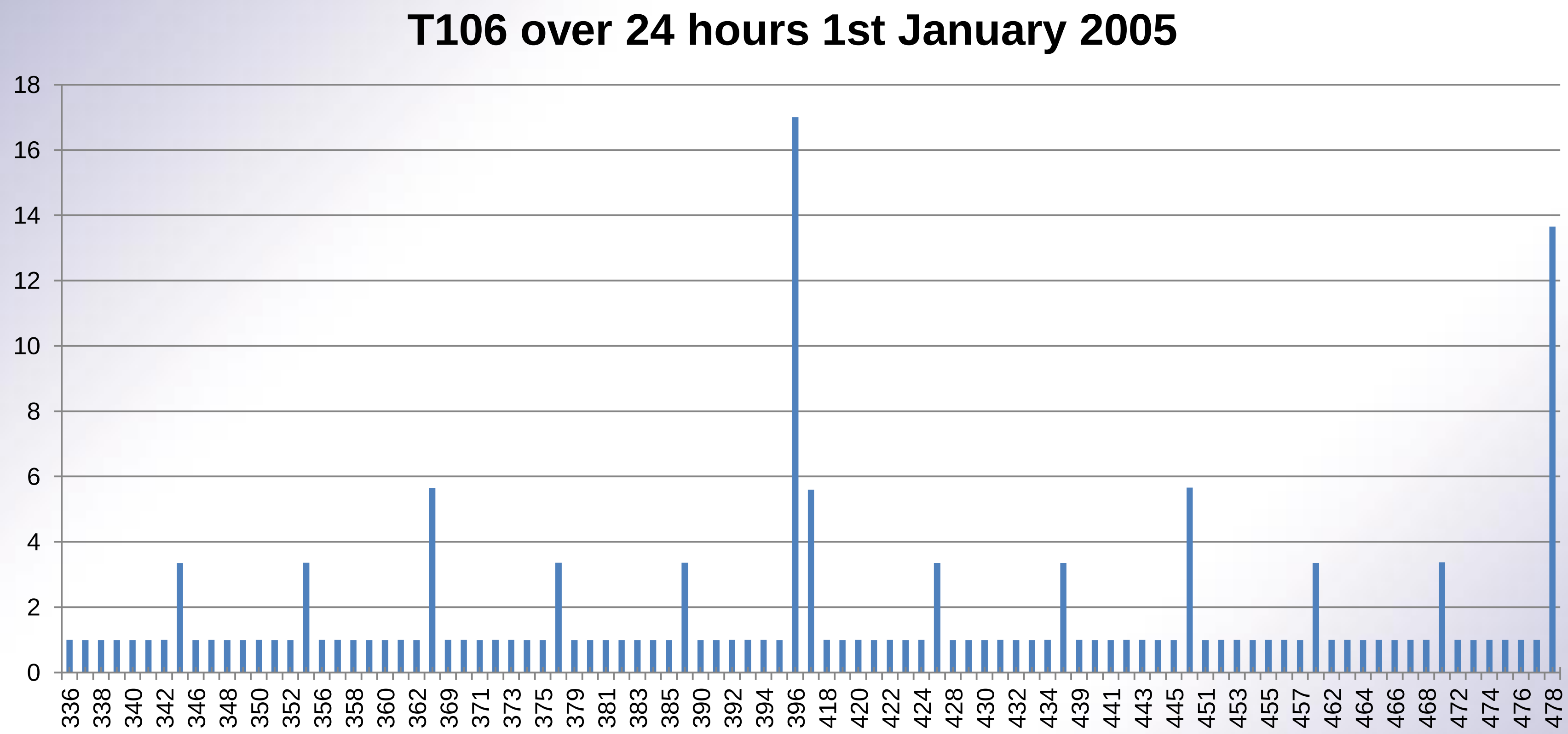
Analysis: File Interaction

- In profiling noticed the blips on certain time steps
  - Initialisation, 2hr, 6hr, 12hr,24hr, end-of-simulation
- Rhythm partly relates to frequency of output
  - 6 hourly read of ECMWF coefficients
  - The PPWRIT and PPREAD of fort.79
  - User specifies frequency of fort.13 GBSTAT reporting
  - User specifies frequency of fort.15 SORZM reporting
- There is a 2hr additional calculation (CALFLU)
- Initial step is huge in comparison to this one day run
  - might be insignificant for decade or even a month run.
  - 350s initial step and 1.0 sec for subsequent 96 iterations

HECToR

RESEARCH COUNCILS UK

Analysis: Per iteration time for T106 on PE80



HECToR

RESEARCH COUNCILS UK

Analysis: scaling to more MPI tasks

Time per iteration for T106 simulation of one day (1st Jan 2005)

MPI	OMP	NPROCI	NPROCK	MYLAT	MYLON	NIV	NBox per patch	Time for initial step	**Time per interval step**	Time for 2 hour step	Time for 6 hour step	Time for 12 hour step	Time for final step
80	T1	5	16	64	10	60	39040	332	**1.00**	3.37	5.7	15.21	13.59
	T2							345	**0.73**	2.18	4.1	17.58	13.87
	T4							359	**0.55**	1.42	3.6	16.08	14.50
160	T1	5	32	64	5	60	19520	327	**0.60**	1.82	3.2	6.57	5.64
	T2							345	**0.49**	1.38	2.9	6.35	5.91
	T4							393	**0.37**	1.00	2.75	6.99	6.33
400	T1	5	80	64	2	60	7808	323	**0.47**	0.94	2.19	7.95	7.02
	T2							338	**0.44**	0.71	2.12	7.3	7.25
	T4							388	**0.36**	0.72	2.19	8.3	7.99

Time in seconds

HECToR

RESEARCH COUNCILS UK

Analysis: Examine runtime profile with CrayPAT

- First pass with a sampling experiment
- Second pass with a tracing experiment
  - Generates a lot of data
  - use the sampler to identify which functions to trace.
  - Additional experiment for IO (-g *sysio,stdio,ffio,aio*)
- Further experiments done using API instrumentation
  - Re-compilation is necessary, intrusive coding
  - pat_record
    - *Selectively turn on logging of data*
  - pat_region
    - *More specific sections of code*
  - Use an iteration monitor to activate logging of data
  - Higher resolution sampling

HECToR

RESEARCH COUNCILS UK

CrayPAT Report (excerpt PE400)

Overall sampling of 96 iterations

% of run	Num samples	
100.0%	28790.9	Total
72.0%	**20717.1**	**MPI**
62.7%	18044.1	mpi_bcast
6.5%	1864.3	MPI_BARRIER
1.3%	383.2	MPI_SENDRECV
23.0%	**6619.3**	**USER**
8.7%	2497.8	consom_
3.8%	1090.9	advy2_
1.8%	513.6	pblscheme_radabs_
1.7%	502.6	advz2_
1.7%	487.5	advx2_
1.5%	425.9	rdemi1x1_
5.1%	**1454.4**	**ETC**

Restrict recording to one iteration (2,3)

% of run	Num samples	
100.0%	**124.1**	**Total**
84.3%	**104.6**	**USER**
42.6%	52.9	consom_
17.4%	21.6	advy2_
8.4%	10.5	advz2_
8.0%	10.0	advx2_
1.8%	2.2	MAIN_
1.3%	1.7	chimie_
13.3%	**16.5**	**MPI**
5.9%	7.3	MPI_SENDRECV
2.8%	3.5	MPI_BARRIER
1.7%	2.1	mpi_recv
1.5%	1.9	MPI_SSEND
1.0%	1.3	mpi_bcast
2.5%	**3.1**	**ETC**

HECToR

RESEARCH COUNCILS UK

Two specific functions investigated

- CONSOM
  - Within a "standard iteration" it accounts for 45% of timing
  - No clear method for improving the time
  - Some improvement in structure
  - Remove conditional
  - Re-order loop index
- CALFLU
  - Only small section where MPI used inefficiently
  - Restructuring did not show significant gain
  - Swamped by MPI_BCAST and an FFT feature

- Next look at two functions dedicated to reporting results

HECToR

RESEARCH COUNCILS UK

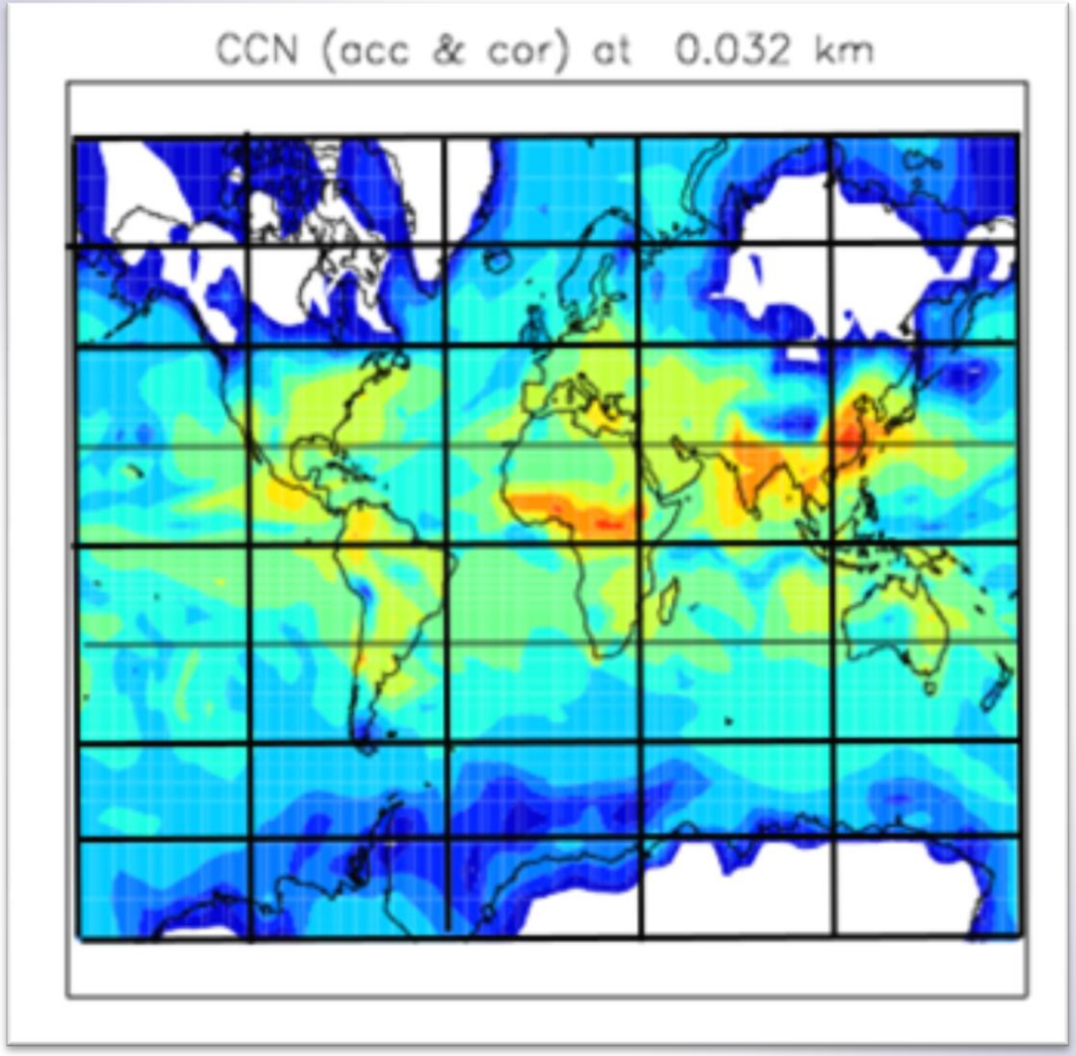
GBSTAT, output information at specific location

- The function extracts a profile of a field as a column of values varying in altitude
- Existing method
  - All data was collected on one MPI task (zero)
  - The task then processed the data
    - *Determine the interpolated value at that altitude*
    - *Each requested field*
  - Columns of data written to fort.13
- Revised method
  - Maintains the interpolation method
    - *now each patch does its own job*
  - First have to locate the ground based stations on the patch
  - Recognise need for halo data
  - Reduces memory requirement
  - Artificially serialise write
    - *so that fort.13 is as previous version*

HECToR

RESEARCH COUNCILS UK

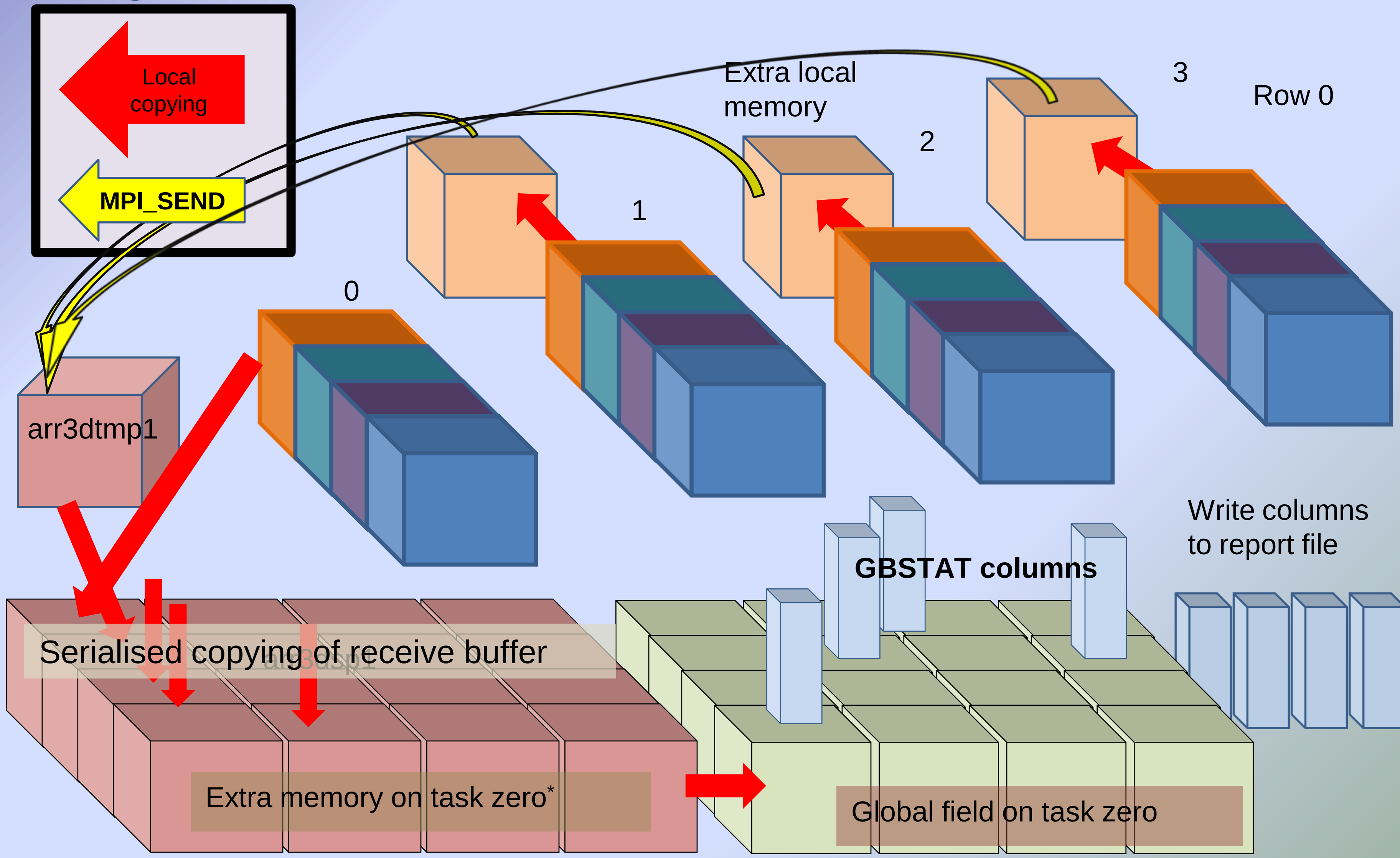
Reminder of domain decomposition



HECToR

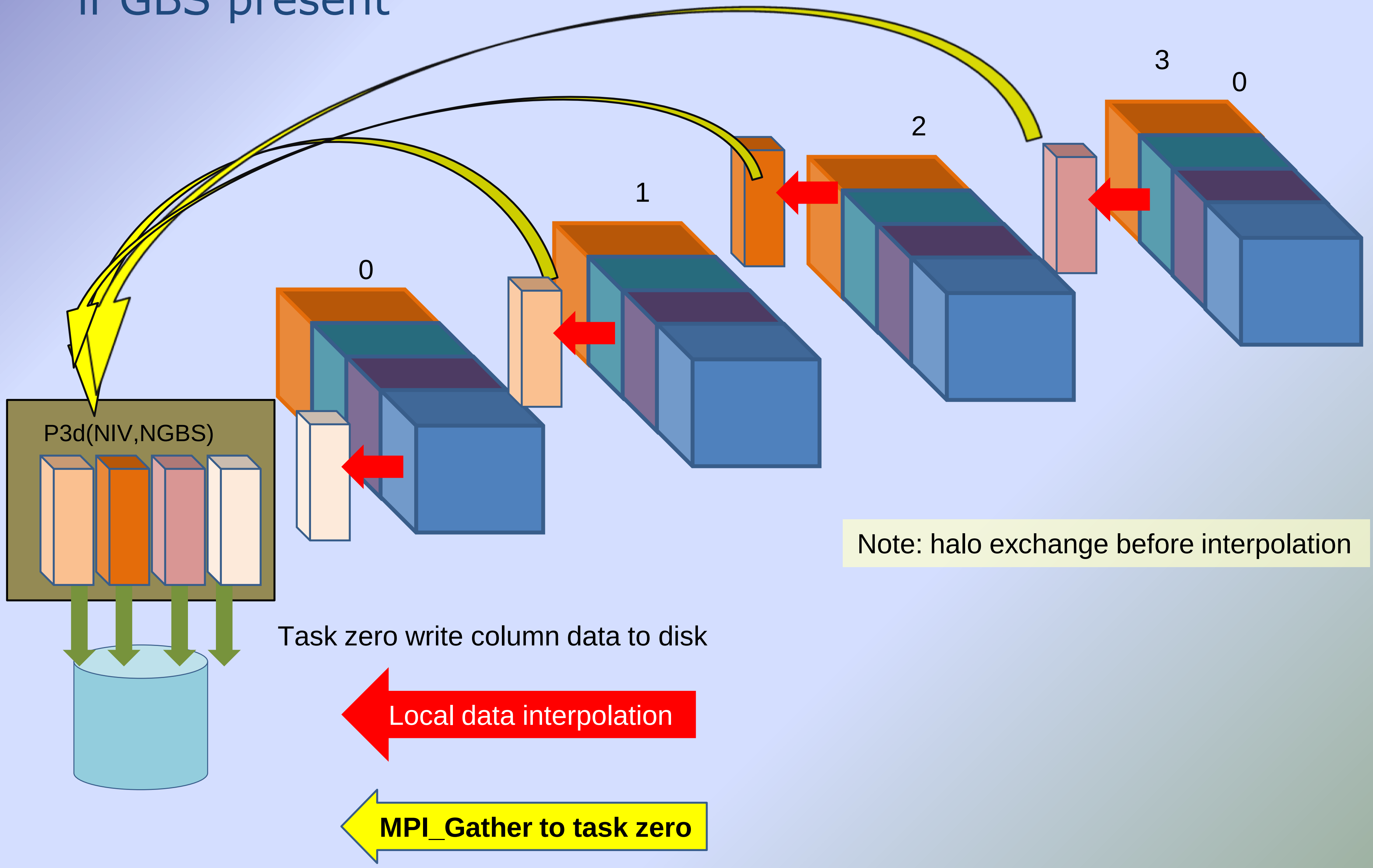
RESEARCH COUNCILS UK

Original GBS Method: task zero does all the work

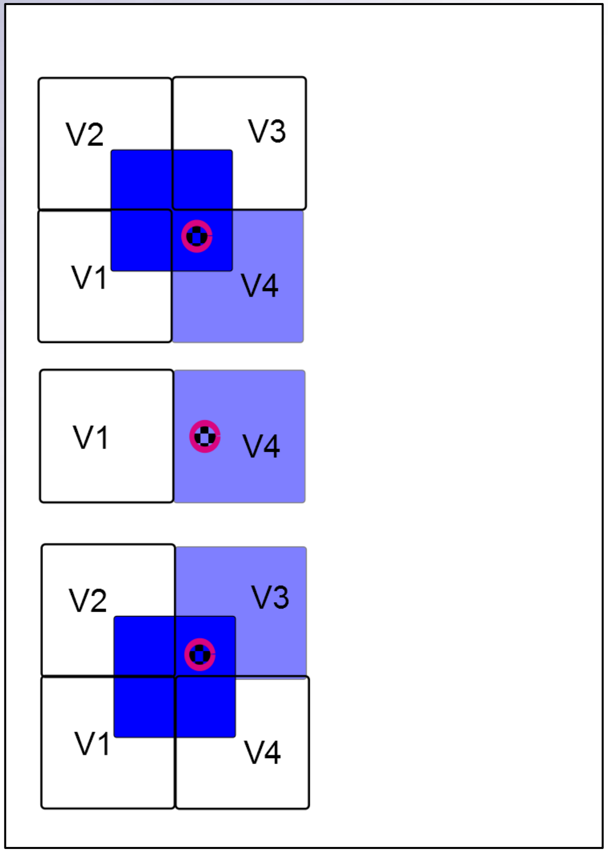


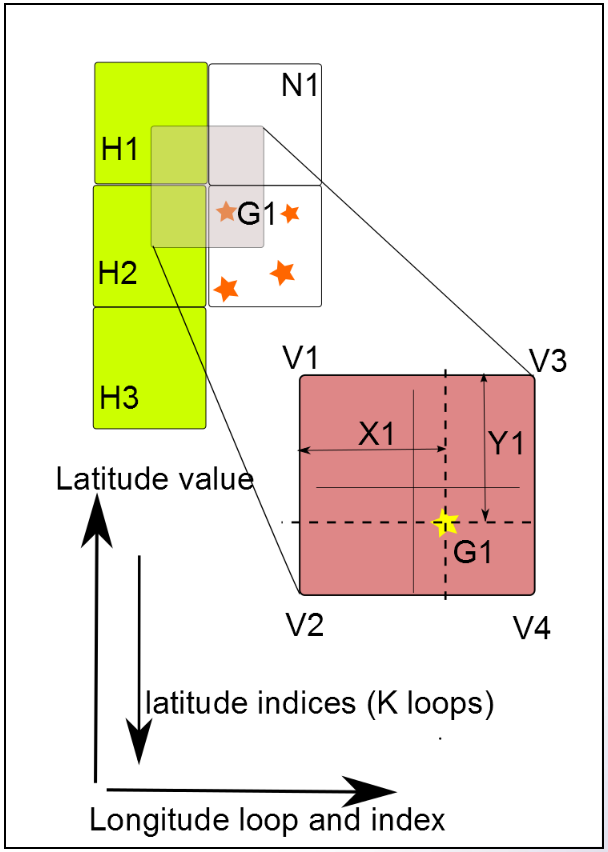
- Task zero has sequentialised the work, repeated for each field
- *The extra memory is statically allocated so all tasks carry it as well

Modified GBSTAT method: each task works if GBS present



GBSTAT interpolation remains in new method





Results of changes to GBSTAT

(1) Estimated memory reduction after removing temporary arrays is 330MB

(2) Reduced amount of data in communication from 265MB to 65MB

(3) Significant reduction in time due to work being done in parallel

TABLE 1: PE80 timing with GBSTAT every second iteration		
Standard GBSTAT	Modified GBSTAT	
Time (seconds)	Time (seconds)	iteration
300.451	307.404	1
9.333	0.981	2
0.983	0.979	3
8.996	0.985	4
0.978	0.986	5
8.789	0.980	6
0.979	0.978	7
8.995	0.983	8
3.362	3.371	9

TABLE 2: PE400 timing with GBSTAT every second iteration		
Standard GBSTAT	Modified GBSTAT	
Time (seconds)	Time (seconds)	iteration
303.692	298.904	1
39.418	0.788	2
0.331	0.424	3
39.366	0.474	4
0.401	0.411	5
39.011	0.459	6
0.391	0.428	7
39.351	0.540	8
0.865	0.889	9

HECToR

RESEARCH COUNCILS UK

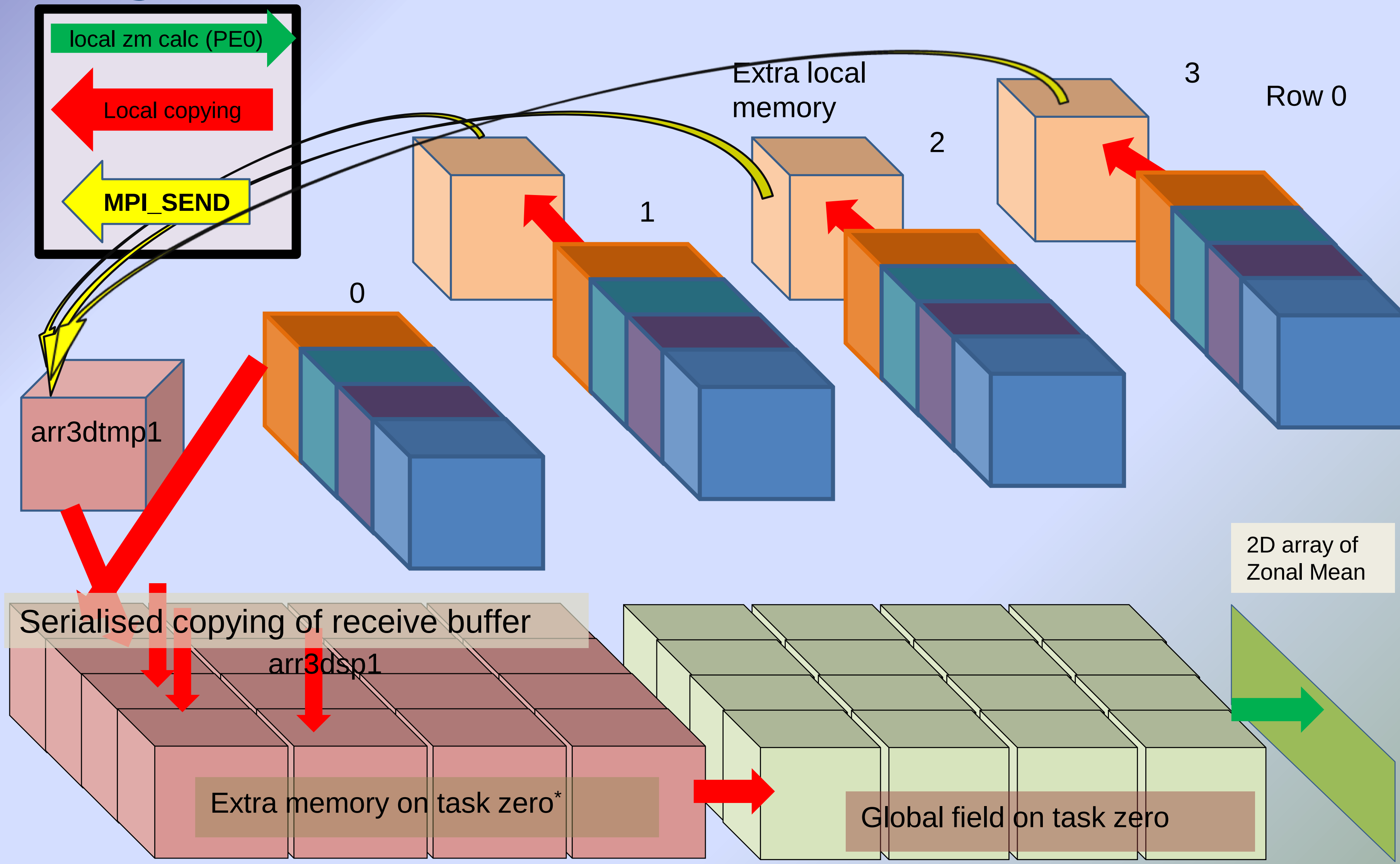
SORZM: for specific field values

- Existing method
  - Collect all field data onto root MPI task
  - Calculate a mean along a latitude
    - *store in a "meridian" plane (LATxNIV)*
  - Serial write to file
- Revised method
  - Calculate mean onto a west most plane (MYLATxNOV)
  - Sum along a row of MPI tasks (to get full longitude sum)
  - stored on end task
  - Divide by LON
  - Gather onto root MPI task (LATxNIV)
  - Serial write to file

HECToR

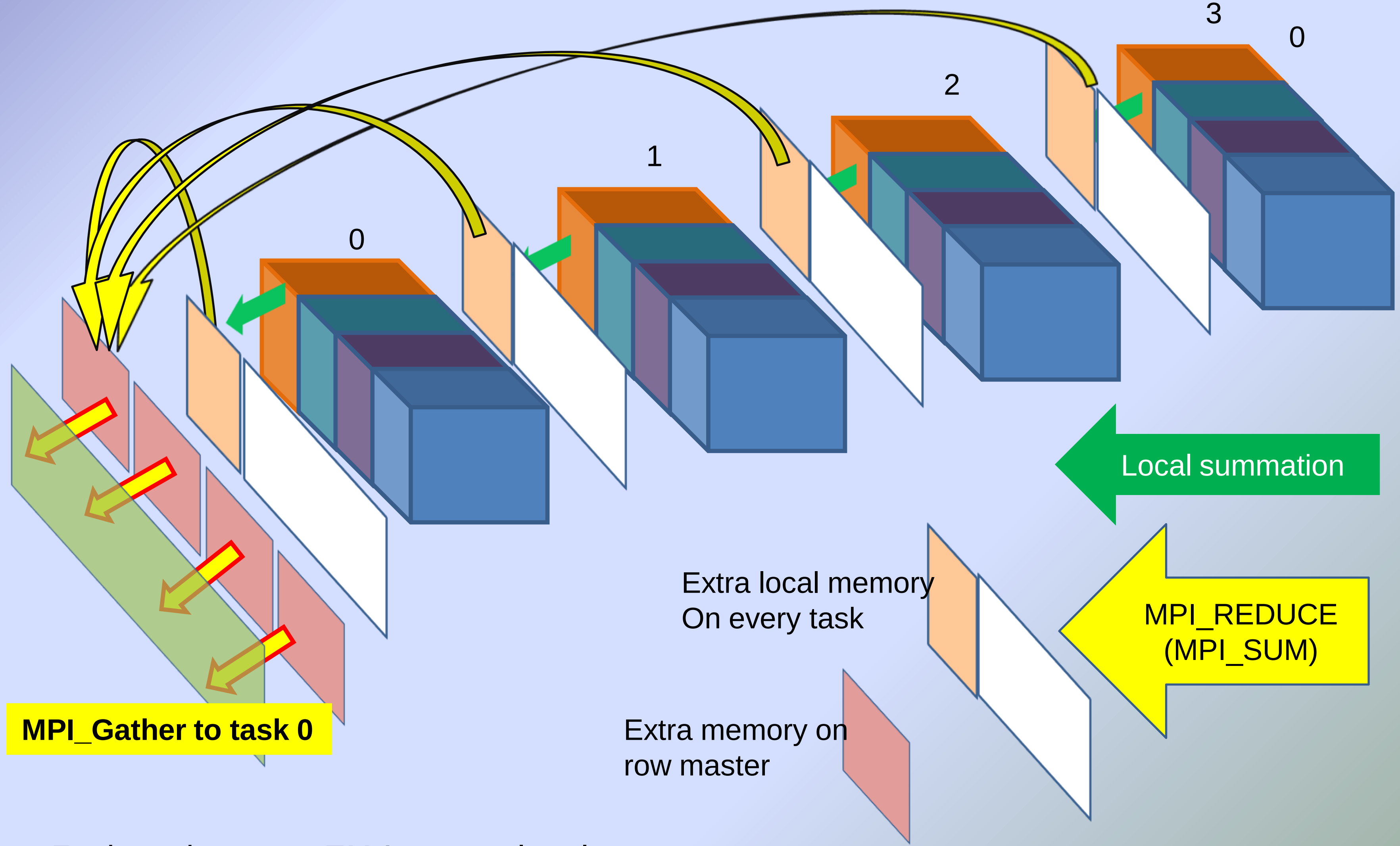
RESEARCH COUNCILS UK

Original SORZM: task zero calculates zonal mean



- Task zero has sequentialised the work
- *The extra memory is statically allocated so all tasks carry it as well

Revised zonal mean calculation



- Each task stores ZM in extra local mem
- Estimated saving of 330MB by removing temporary arrays

Results of changed SORZM

(1) Reduced amount of memory in subroutine 330MB

(2) Reduction in communicated data by 278MB, but replace with 4MB of communication

(3) Significant reduction in time for the step

Table 3: PE 80 , effect of changed SORZM			
Normal Run	Standard SORZM	Modified SORZM	iteration
384.580	377.155	385.091	1
0.531	4.643	0.545	2
0.531	4.649	0.545	3
0.527	4.659	0.541	4
0.525	4.659	0.546	5
0.528	4.651	0.544	6
0.528	4.656	0.543	7
0.528	4.662	0.550	8
1.496	5.602	1.509	9
0.531	4.618	0.550	10

Table 4 : PE400, effect of modified SORZM			
Normal run	Standard SORZM	Modified SORZM	iteration
375.762	415.400	**396.894**	1
0.343	41.554	**0.351**	2
0.333	41.582	**0.327**	3
0.322	41.728	**0.346**	4
0.330	41.193	**0.376**	5
0.327	41.762	**0.331**	6
0.339	41.557	**0.335**	7
0.335	41.600	**0.343**	8
0.656	41.774	**0.658**	9
0.335	41.738	**0.332**	10

HECToR

Enforced activation of SORZM every step to demonstrate effect

RESEARCH COUNCILS UK

Feel good factor

- The changes to GBSTAT and SORZM have allowed researchers to see these as less expensive and are free to do investigations
- Developers have seen the opportunity to re-use the GBSTAT for satellite analysis (dynamic form)
  - Orbit crosses terminator twice per day indifferent locations
- Now they are asking further questions on code refactoring

HECToR

RESEARCH COUNCILS UK

Review PPREAD

- Existing method
  - Flag to say if the data has space for halo storage
    - *Has a conditional test of the flag*
  - Subsequent serial read of a plane of global data
  - Copied into a specific buffer location
    - *Per-process send of sub-section of 2d array*
  - Copy into local data structure
- Revised method
  - Call a new function with a data type
    - *"with-halo" or "no-halo"*
  - Serial read of a plane of global data
  - Use MPI_Scatterv; using the custom Datatype
    - *Let MPI do the packing and unpacking.*

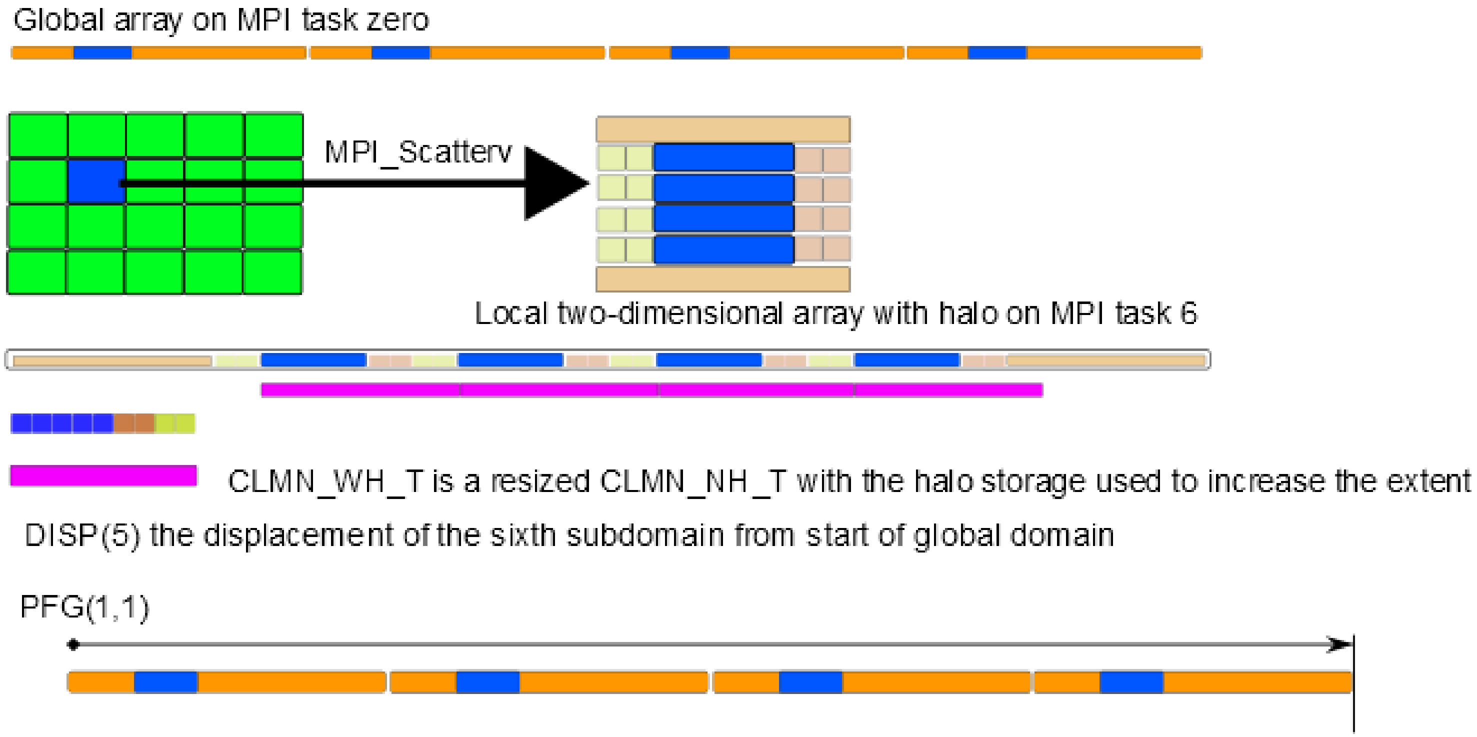
Review PPWRIT

- Existing method
  - Packing a local buffer with a sub-plane of data
  - Send to task zero (or nominated ROOT)
  - Receiving on ROOT from each MPI task in turn
  - Unpack sub-plane into global locations
  - Write global plane to Fortran unformatted sequential file
- Revised method
  - Call a new function with a datatype
    - *"with-halo" or "no-halo"*
  - Use custom datatypes
  - Use MPI_Gatherv with appropriate datatype
  - Write global plane to Fortran unformatted sequential file

HECToR

RESEARCH COUNCILS UK

New data structures introduced




```
CALL MPI_Gatherv(PFL(1,1),NSND,CLMN_WH_T,PFG(1,1),NRCV,DISP,BLK_NH_T, ROOT,UCOMM,ERR_MPI)
```

Outcome

- New data structures
  - Code is neater
  - Easier to maintain
  - Easier to extend to other areas
- Interface now looks like

```
CALL PPREAD(IFRD, S0 (NIMN,NKMN,L,JV), .TRUE., 0)

CALL PPRD (IFRD,NIMN,NIMX,NKMN,NKMX,CLMN_WH_T,S0(NIMN,NKMN,L,JV))
```

- Currently unclear any performance gain
- Swamped by broadcast and other work in the section of code.

HECToR

RESEARCH COUNCILS UK

Review NetCDF

- The "write_cdf" routine is actually "write fort.9"
  - It does too much additional processing
- There is typically a collection of data to the root task (0) followed by a call to

  unitom_write_var a wrapper for nf90_put_var()
- A choice is available
  - Could replace "coll" with MPI_Gather
    - *Will use the new data types that define the data structures*
  - Potential to use HDF5 parallel enabled NetCDF
    - *Will have to remove all the "if (myproc.eq.0)" filters*

HECToR

RESEARCH COUNCILS UK

Summary of this work

- Several hotspots have been targeted
  - Seen gains from revision of
    - *GBSTAT*
    - *SORZM*
  - Not so clear with PPREAD (yet)
  - PPWRIT will be used further with NetCDF files
- Further gains could be made in “hot” routines
  - If more time available
- Additional feedback in the form of advice and observations

Acknowledgements

- Professor Martyn Chipperfield for weekly discussion
- University of Leeds, for the loan of office space
- RCUK the HECToR DCSE programme
- NAG
- My colleagues at NAG Manchester
- The HECToR help desk and support team in Edinburgh

Thank you for your attention!

HECToR

RESEARCH COUNCILS UK