IFS Scalability and Computational Efficiency

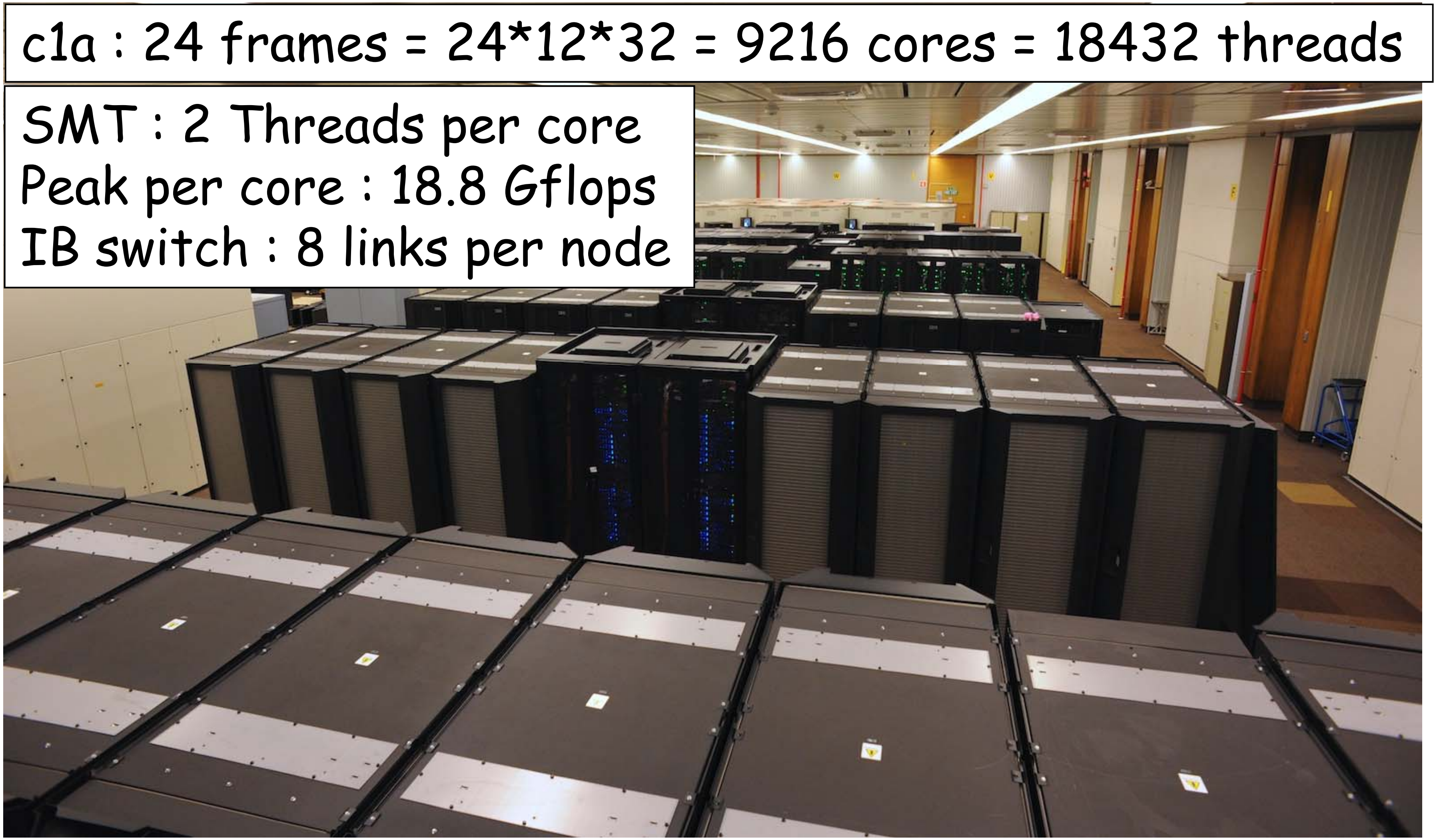
Deborah Salmond & Mats Hamrud

ECMWF

One of ECMWF's two IBM Power6 clusters

c1a : 24 frames

One of ECMWF's two IBM Power6 clusters



c1a : 24 frames = 24*12*32 = 9216 cores = 18432 threads

SMT : 2 Threads per core
Peak per core : 18.8 Gflops
IB switch : 8 links per node

Plan of talk

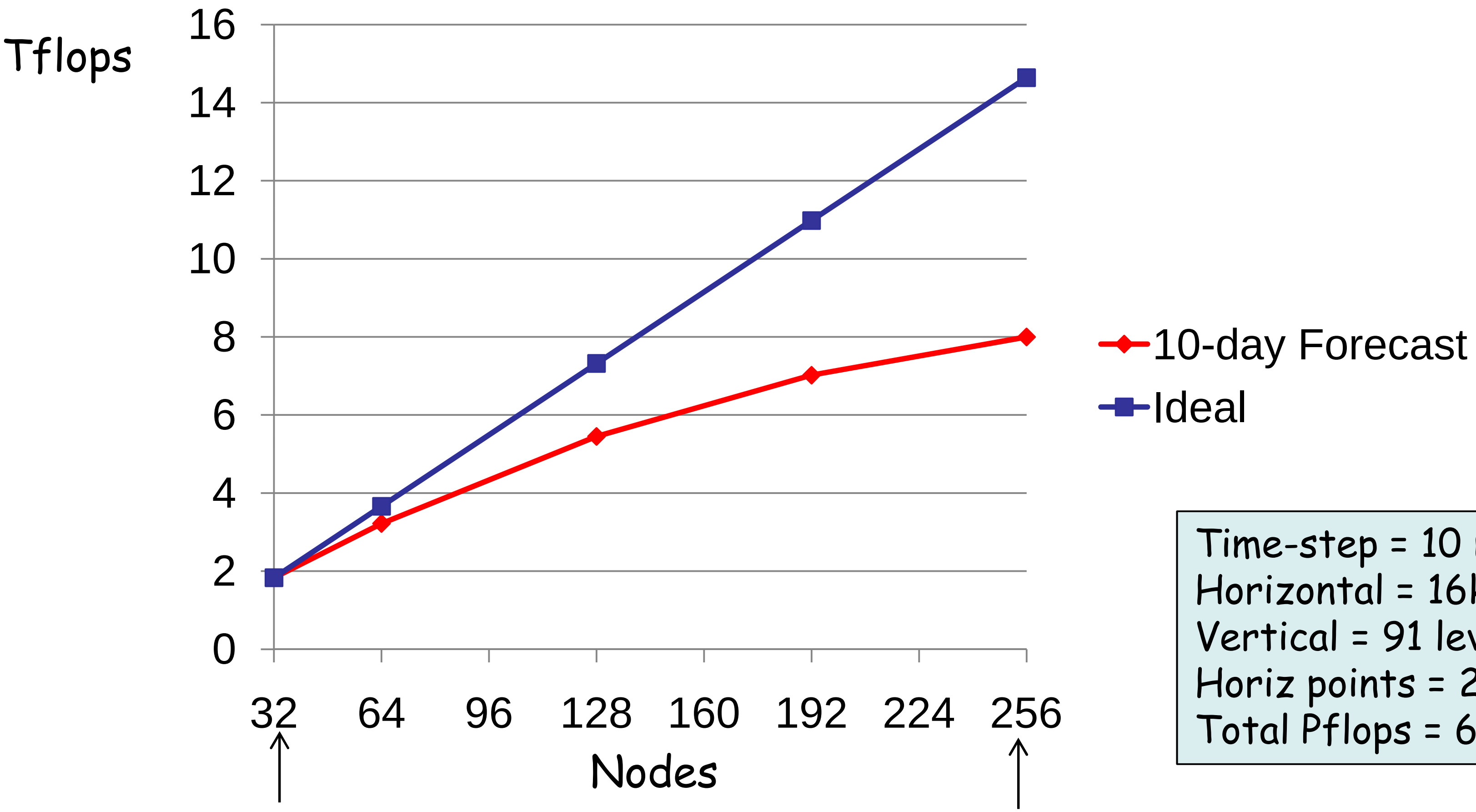
- IFS 10-day forecast and 4D-Var
  - Scalability & Computational efficiency
  - Comparison of Forecast and 4D-Var
  - Profiles of different parts of 4D-Var
  - Study of I/O scalability
  - Recent Optimisations and Scalability Improvements

- Plan to improve scalability of 4D-Var

4-point plan to improve scalability of IFS

- Analysis
- Short term
  - Technical improvements in scaling in the current IFS
- Medium term
  - Major restructuring of 4D-Var code
- Longer term
  - Algorithmic changes

T1279 Forecast runs up to whole cluster

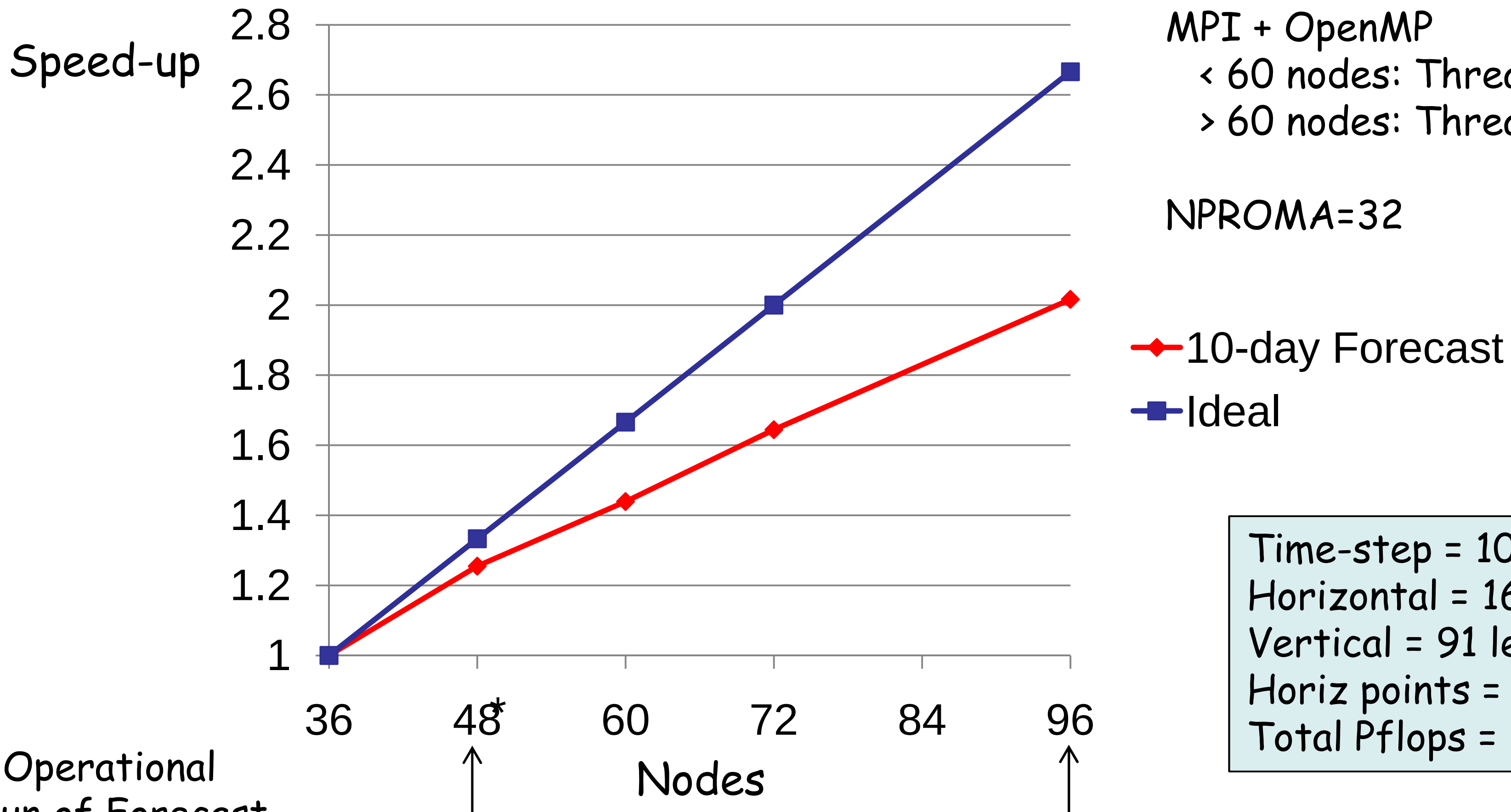


9% peak
2048 threads

5% peak
16384 threads

Time-step = 10 min
Horizontal = 16km
Vertical = 91 levels
Horiz points = 2x10^6
Total Pflops = 6.5

Speed-up of T1279 Forecast



MPI + OpenMP
< 60 nodes: Threads=8
> 60 nodes: Threads=16

NPROMA=32

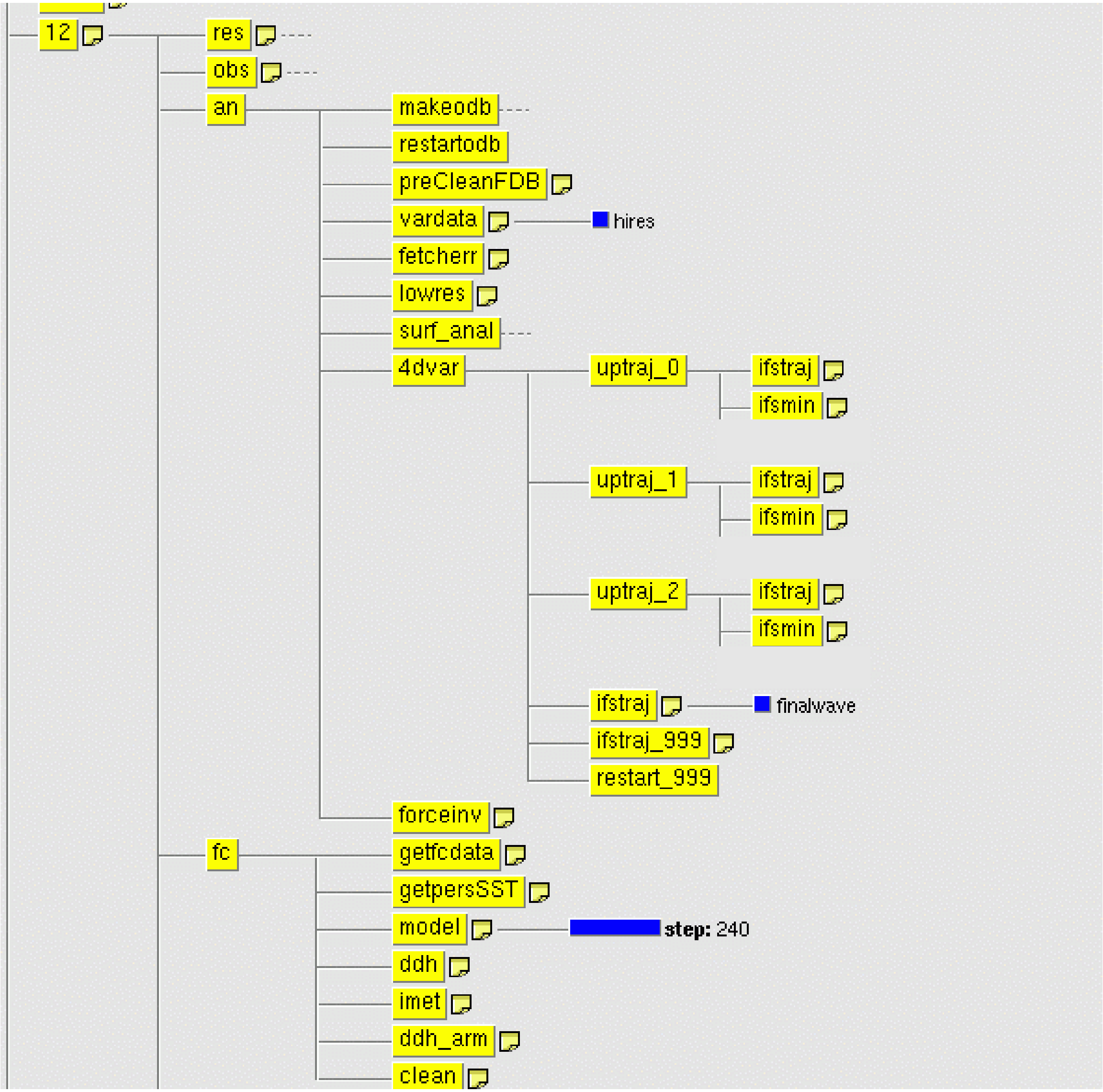
Time-step = 10 min
Horizontal = 16km
Vertical = 91 levels
Horiz points = $2x10^6$
Total Pflops = 6.5

*Operational run of Forecast and 4D-Var on 48 nodes

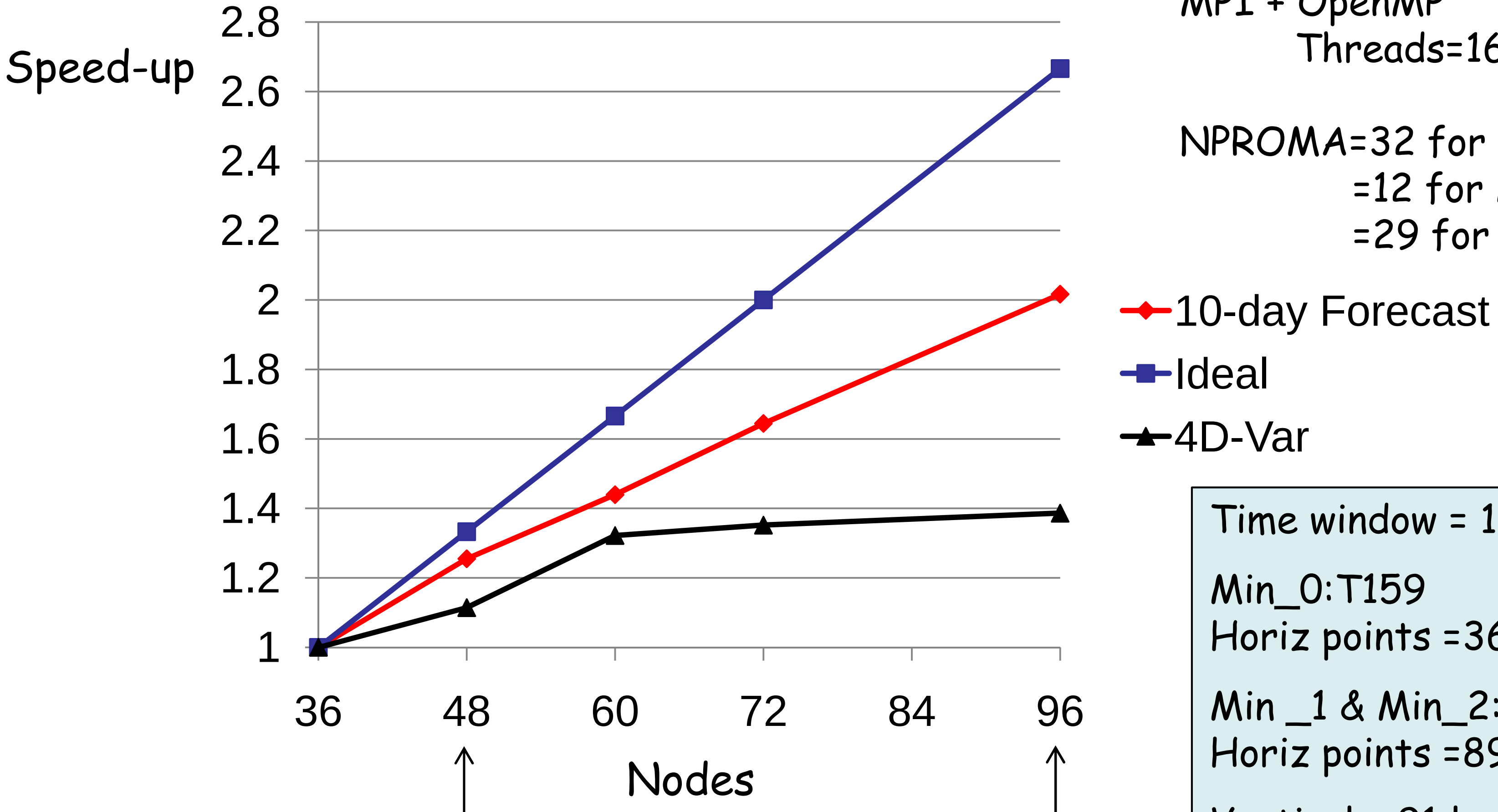
2.3 Tflops
8.1% peak
3072 threads

3.7 Tflops
6.5% peak
6144 threads

4D-Var and 10-day forecast



Speed-up of T1279 4D-Var



MPI + OpenMP
Threads=16

NPROMA=32 for Traj
=12 for Min_0
=29 for Min_1

↑ 0.83 Tflops
2.9% peak
3072 threads

↑ 1.03 Tflops
1.8% peak
6144 threads

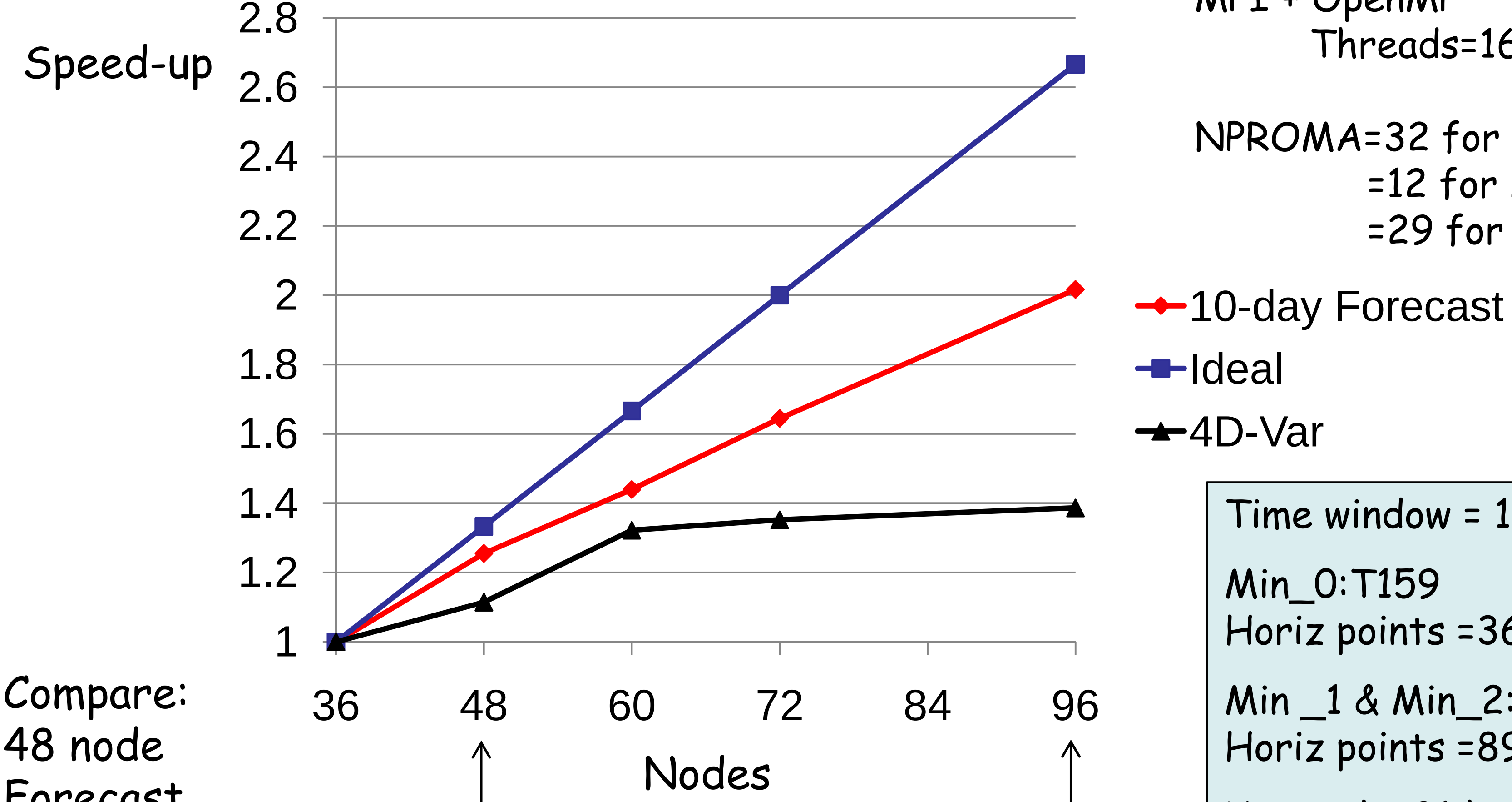
Time window = 12hr

Min_0:T159
Horiz points =36000

Min _1 & Min_2:T255
Horiz points =89000

Vertical = 91 levels
Total Pflops = 2.3

Speed-up of T1279 4D-Var



MPI + OpenMP
Threads=16

NPROMA=32 for Traj
=12 for Min_0
=29 for Min_1

Time window = 12hr

Min_0:T159
Horiz points =36000

Min _1 & Min_2:T255
Horiz points =89000

Vertical = 91 levels
Total Pflops = 2.3

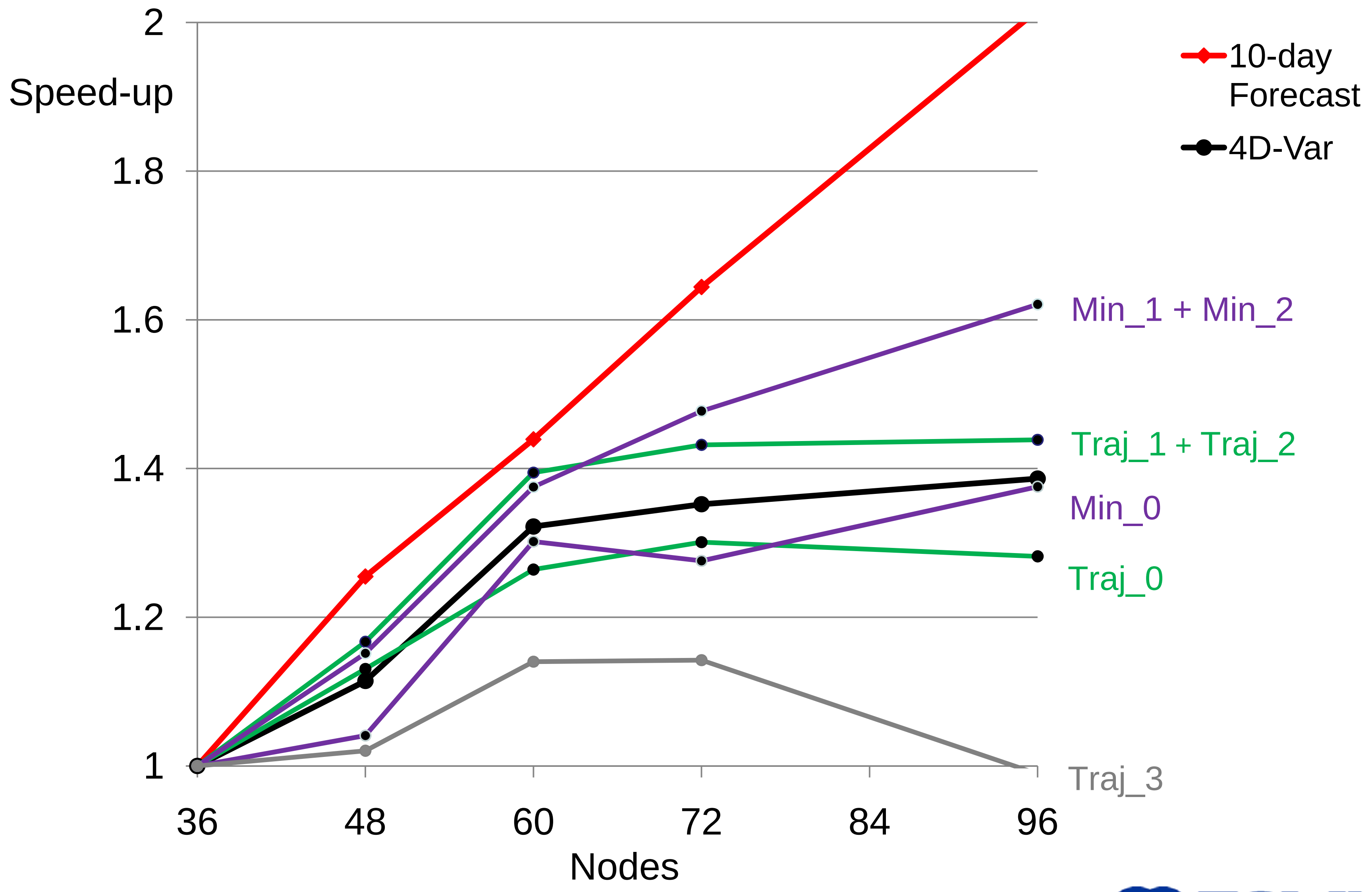
Compare:
48 node
Forecast
2.3 Tflops
8.1% peak

↑
0.83 Tflops
2.9% peak
3072 threads

↑
1.03 Tflops
1.8% peak
6144 threads

ECMWF

Speed-up of Different parts of 4D-Var



Computational efficiency of T1279 4D-Var & 10-day Forecast on 48 nodes

Step	WALLTIME in seconds	%peak
Traj_0	395	3.1
Min_0 (T159)	540	1.5
Traj_1	261	4.5
Min_1 (T255)	495	2.7
Traj_2	282	4.3
Min_2 (T255)	449	2.8
Traj_3	430	2.9
4D-Var -Total	2854	2.9
10 day Forecast	2825	8.1

Computational efficiency of T1279 4D-Var & 10-day Forecast on 48 nodes

Step	WALLTIME in seconds	%peak	Description
Traj_0	395	3.1	T1279 : I/O – full obs
Min_0 (T159)	540	1.5	T159 : 70 iterations
Traj_1	261	4.5	T1279 : 72 time steps
Min_1 (T255)	495	2.7	T255 : 25 iterations
Traj_2	282	4.3	T1279 : 72 time steps
Min_2 (T255)	449	2.8	T255 : 25 iterations
Traj_3	430	2.9	T1279 : I/O – full obs
4D-Var -Total	2854	2.9	
10 day Forecast	2825	8.1	T1279 : 1440 time steps

Top 10 routines from Xprofiler and pmapi

10-day Forecast

```
 %time   name           Mflops

  5.4   .datb13c         6132
  4.5   .cloudsc_         718
  3.9   .laitri_         1299
  2.8   .lascaw_          147
  2.2   .srtm_spcvrt_     782
  2.2   _exp
  2.1   .vdfmain_         740
  1.9   .laitli_         1035
  1.8   .cloudvar_        448
  1.8   .srtm_reftra_     600
  1.8   .cuadjtq_        1168
  1.8   .radlswr_         223
```

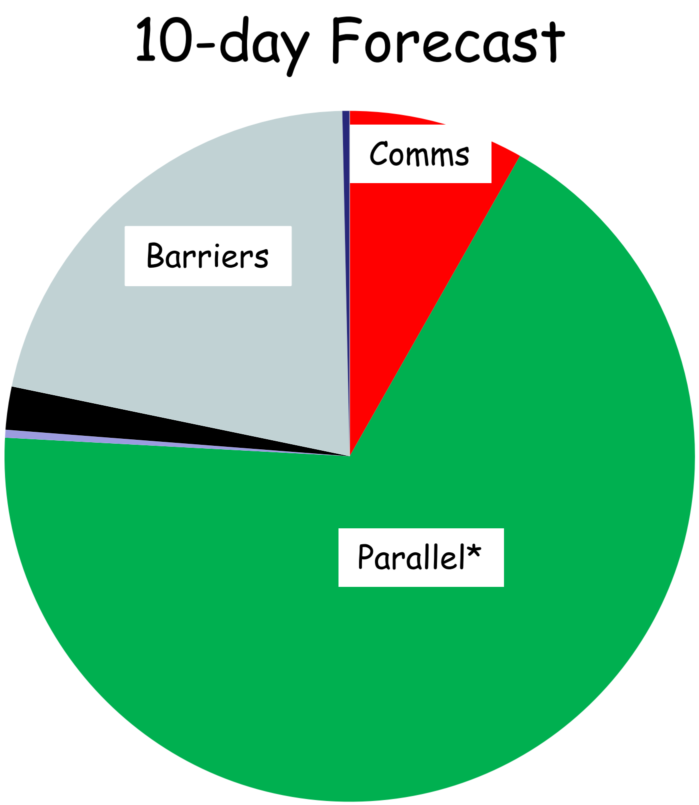
Min_2

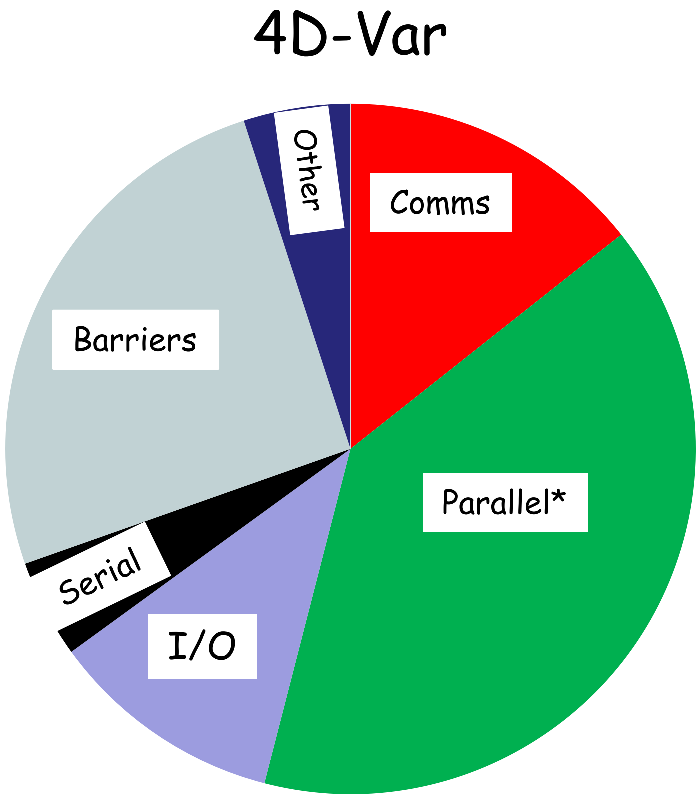
```
 %time   name            Mflops

  7.3   .lwvdrad_          383
  6.1   .cloudstad_        443
  5.5   .lwvdrtl_          651
  5.1   .lwvdr_            357
  3.7   .lwcad_            417
  2.9   .cloudsttl_        554
  2.1   _exp
  2.0   .lwctl_            652
  1.7   ._stripe_hal_pkts
  1.6   pow
  1.3   .swniad_           314
  1.1   .datb13c          2672
```

GSTATS

- Timing around significant parts of the IFS code
- Classify as
  1. PARALLEL = OpenMP parallel sections
  2. SERIAL = non-OpenMP
  3. COMMS = MPI communications
  4. I/O = I/O + 'I/O support'
  5. BARRIERS
  6. OTHER
- Runs with extra barriers put around communications so barriers time is artificially high
  - Part of the barrier time comes from jitter – expect this to reduce on P7
- Runs not dedicated - but used co-scheduler

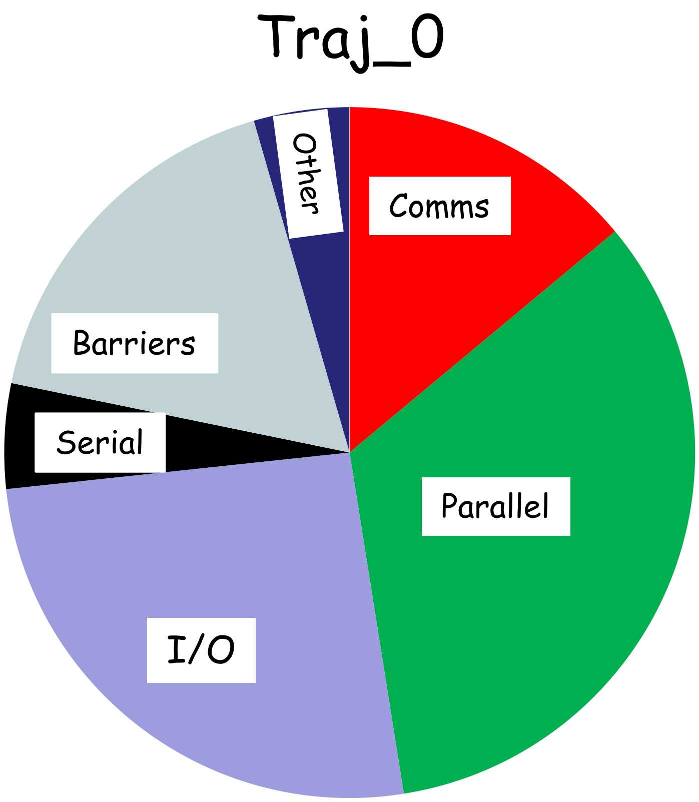
GSTATS for T1279 4D-Var & Forecast on 48 Nodes

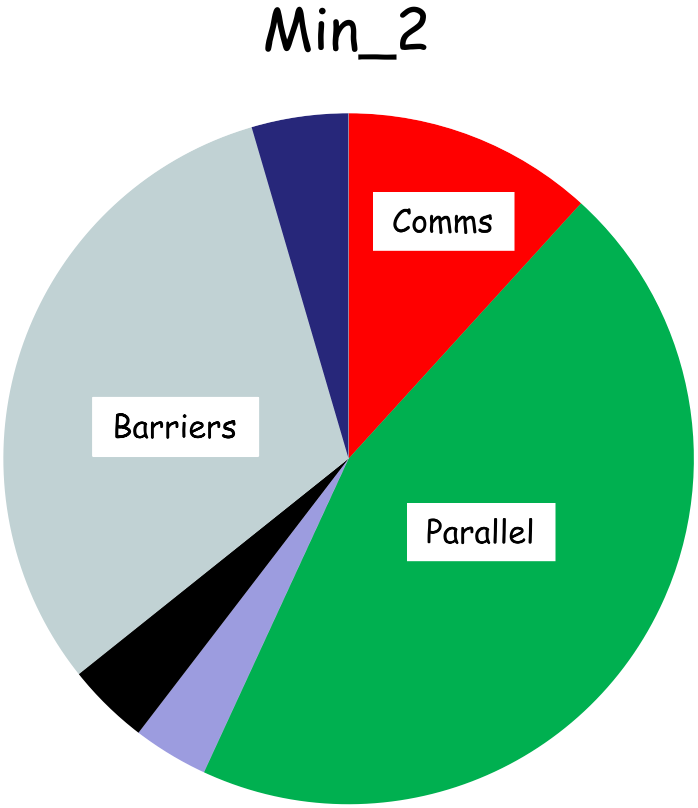




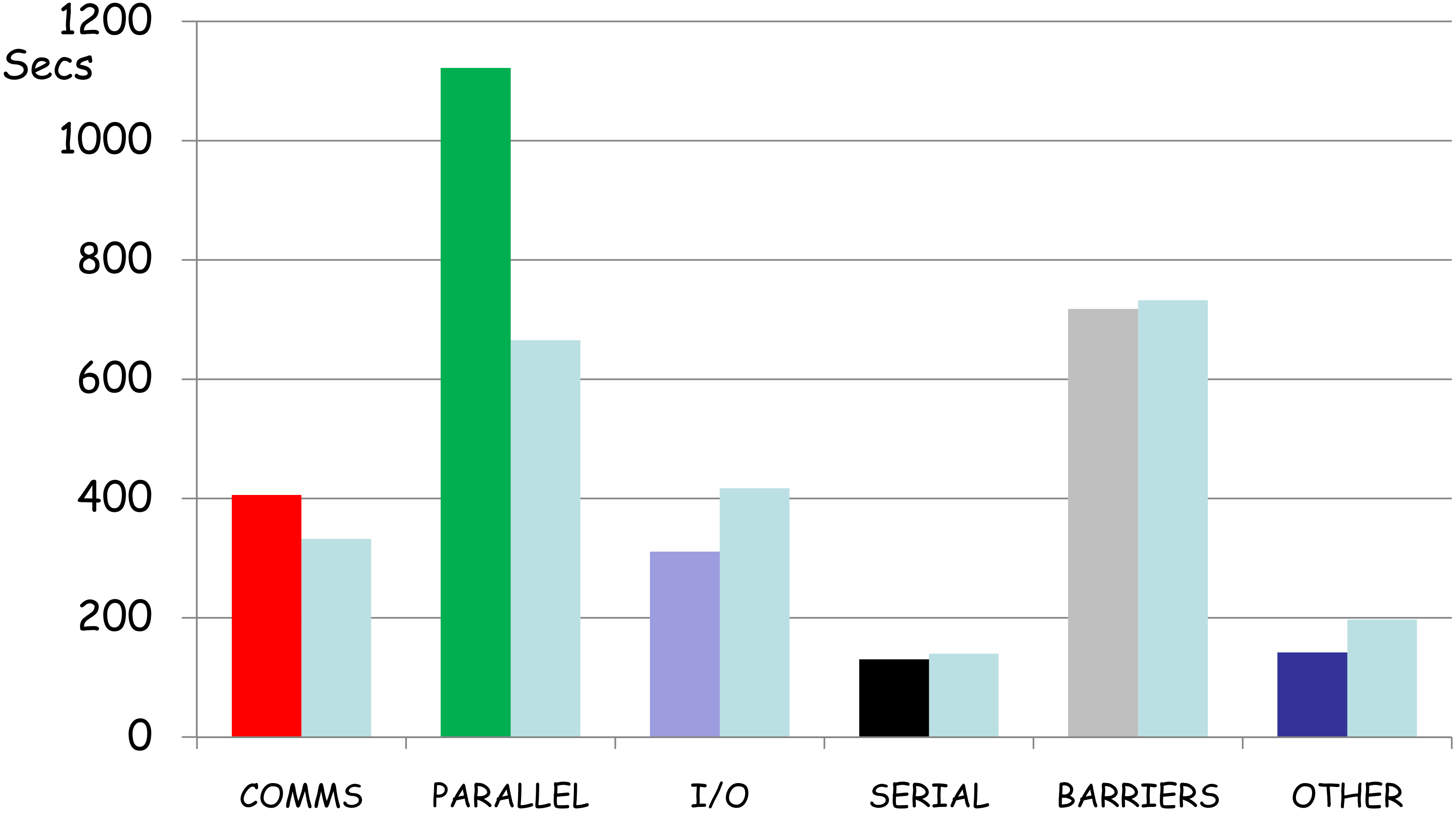
*Parallel is the part that scales well and has best Mflops - including Legendre transform (5% of total for Forecast & 2% of total for 4D-Var)

GSTATS for 4D-Var sub-tasks on 48 nodes

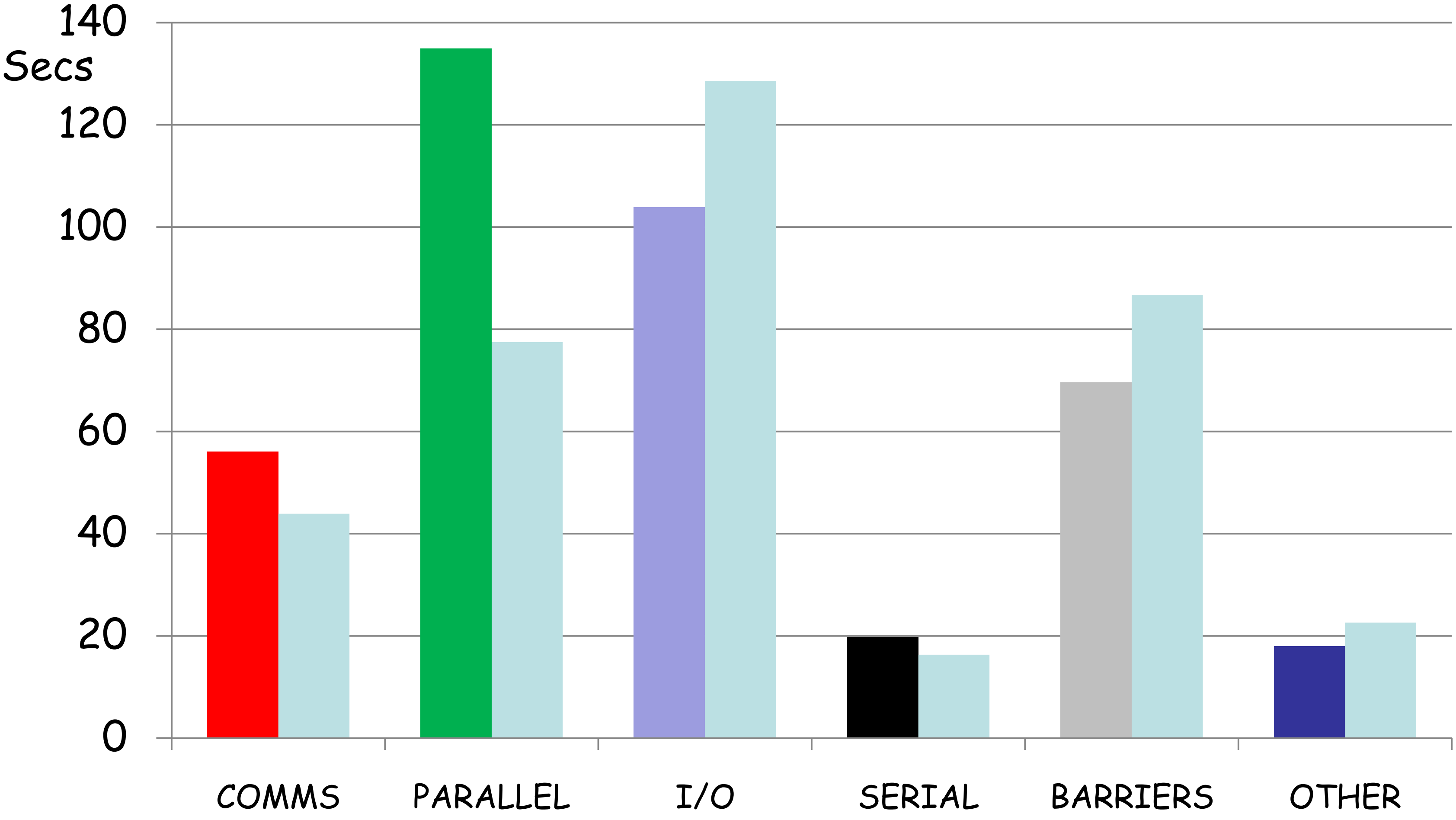




Scalability of 4D-Var: 48 to 96 nodes



Scalability of Traj_0: 48 to 96 nodes



Study of I/O scalability

- Initial conditions(PE-0 and broadcast), ODB(parallel), FDB(asynchronous), internal files between steps.

- GSTATS for ODB - Traj_0

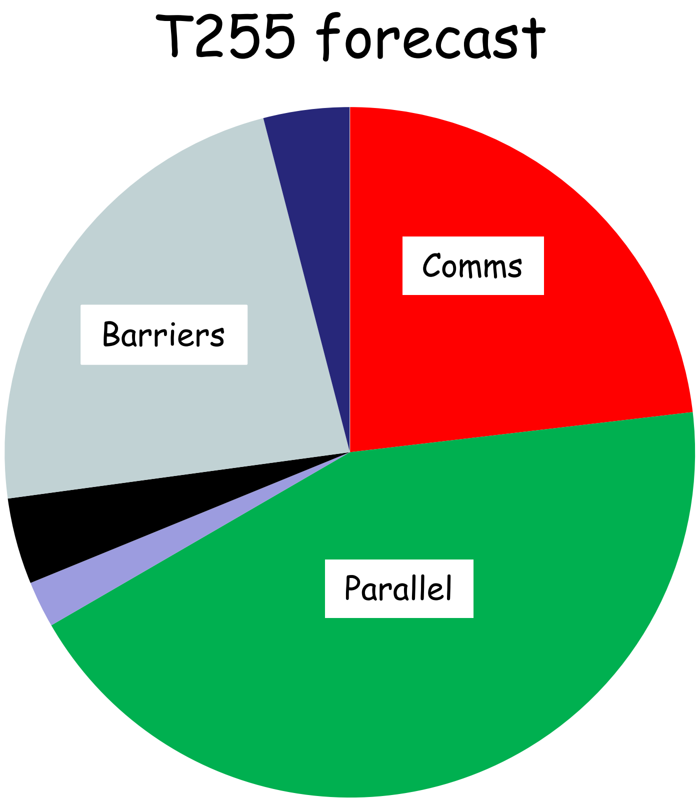
		48 node	96 nodes	Speed-up	Secs lost
1791	IO- DB in READOBA	17.1	27.4	0.62	18.8
1792	IO- DB in WRITEOBA	40.4	41.2	0.98	21.0

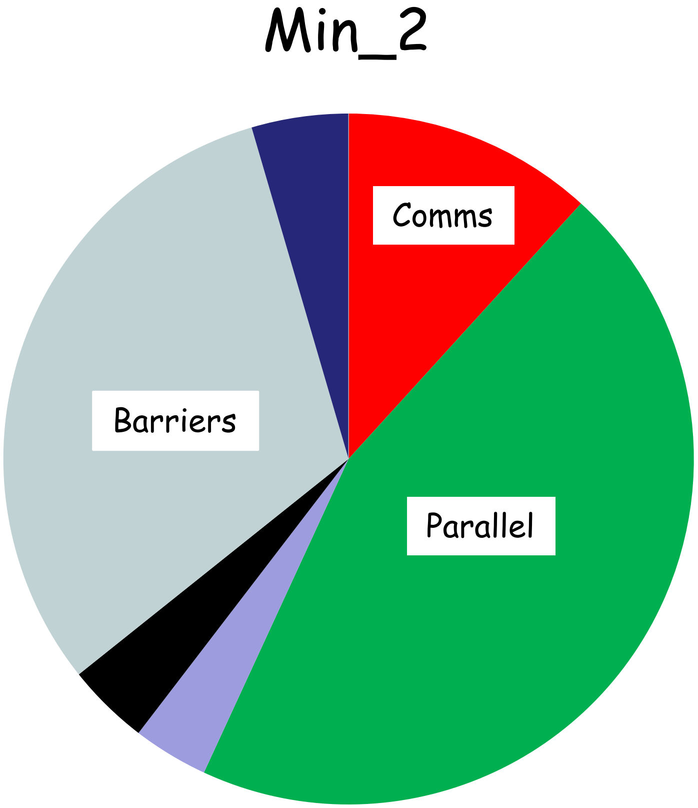
  - I/O and comms related to I/O

- JIO for Initial conditions - Traj_0

Nodes	MSEC	MB	RATE	CALLS	File
48	2291.2	2417.2	1055.0	26793	ICMGGINIUA
96	3520.3	2417.2	686.7	26793	ICMGGINIUA

GSTATS for 48 Node runs of Min(T255) & T255 forecast with same number of timesteps

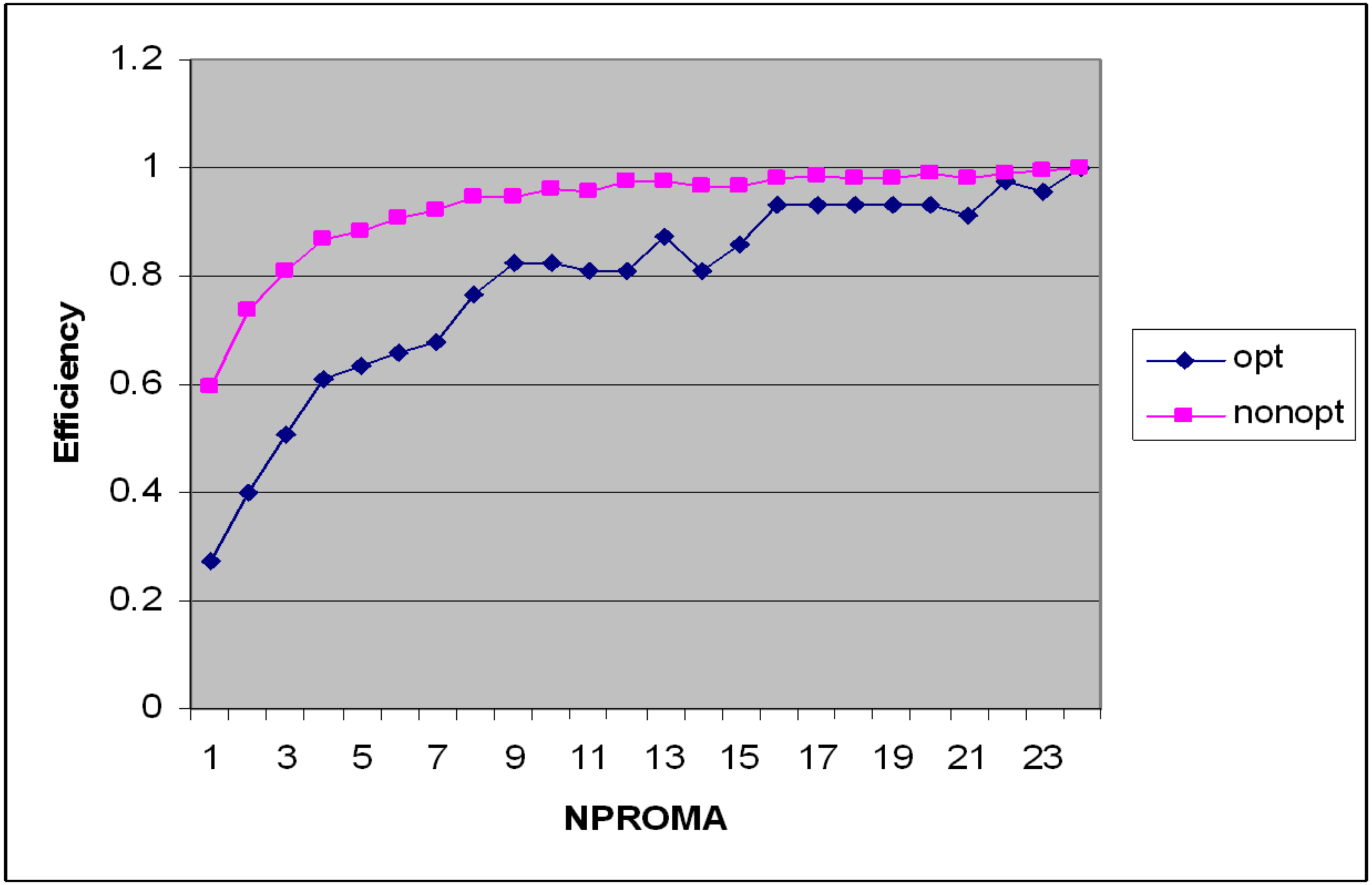




Look for more Scalability

- Use of profilers to analyse performance
  - Xprofiler, Dr.Hook, HPM, GSTATS, JIO
- Reduce time for scripts - now takes 12% of total
- More Parallelism
  - Higher resolution
    - ECMWF strategy
      - More vertical levels in 2011
      - Horizontal resolution to 10km in 2014/15
  - Reduce NPROMA?
- Reduce barrier time
  - 50% comes from jitter
  - 50% from Load-imbalance

NPROMA



Efficiency
1.2
1
0.8
0.6
0.4
0.2
0
1
3
5
7
9
11
13
15
17
19
21
23
NPROMA
opt
nonopt

Recent Optimisations and Scalability improvements-1

36R1

Message passing optimisation of DDH (Forecast)	1-10%
Optimisations for IBM Power6	10%
Optimisation of Operational Post-processing (Forecast)	20%
Optimisations of TL/AD Radiation and Dynamics	2%

36R2

Script optimisation	8%
Improve parallelisation of Control Vector Dot-Product	1%
Parallelise distribution of spectral fields for read of spectral data	1%
OpenMP for distribution of vertical correlation matrices for wavelet Jb	1%
MPL_ALLREDUCE function changed to use a binary tree construct	1%
Optimisation of SL Comms for 4D-Var Minimisation	3%
Improve flexibility in partitioning of spectral space	1%
Improvements to message passing in Rain Assimilation	2%
Speed-up bufr2odb jobs	
Optimisation of LW radiation	1%
Improvement of parallelism for control vector I/O	1%
Improve scalability of the implicit Coriolis solver	1%

Recent Optimisations and Scalability improvements-2

36R3

Move Rttov9 allocations to higher level	1%
OpenMP Parallelisation of Snow analysis	
Redistribute ODB for All-sky data	2%
Optimisation of ODB MPI Communications	1%

36R4

Optimisation of UPDTIM (remove copies)	1%
Optimisation of "here documents“ (scripts)	
Optimisation of new CLOUDSC	1%
Optimisation of TL/AD Physics	1%
VarBC order independent sums	
Optimise reading of RTTOV coefficient files	3%
Optimisation of LASCAWTL/AD (copies at subroutine call)	1%
Load-Balancing of Bufr2ODB	
Optimisation of ODB message passing	1%

Top 10 routines from Xprofiler and pmapi

10-day Forecast

%time	name	Mflops
5.4	**.datb13c**	**6132**
4.5	.cloudsc_	718
3.9	.laitri_	1299
2.8	.lascaw_	147
2.2	.srtm_spcvrt_	782
2.2	_exp	
2.1	.vdfmain_	740
1.9	.laitli_	1035
1.8	.cloudvar_	448
1.8	.srtm_reftra_	600
1.8	.cuadjtq_	1168
1.8	.radlswr_	223

Min_2

%time	name	Mflops	
7.3	.lwvdrad_	383	*445
6.1	.cloudstad_	443	
5.5	.lwvdrtl_	651	
5.1	.lwvdr_	357	
3.7	.lwcad_	417	
2.9	.cloudsttl_	554	
2.1	_exp		
2.0	.lwctl_	652	
1.7	._stripe_hal_pkts		
1.6	pow		
1.3	.swniad_	314	
1.1	**.datb13c**	**2672**	

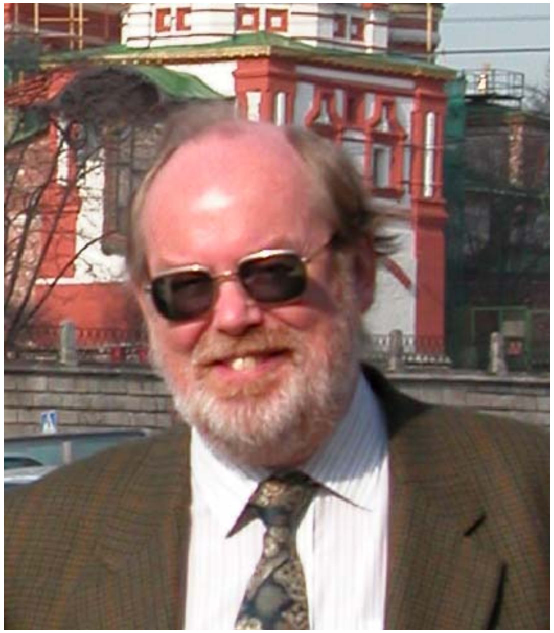
* Loops re-ordered to get better use of streaming from memory

4-point plan to improve scalability

- Analysis
  - Scalability Project
    - Better understanding of opportunities to improve scalability of 4D-Var
    - Report from the IFS Scalability Project: Mats Hamrud

    http://www.ecmwf.int/publications/library/do/references/list/14#2010
- Short term
  - Technical improvements in scaling within the current IFS
- Medium term
  - Major restructuring of 4D-Var code
    - Run 4D-Var as a single execution
- Longer term
  - 4D-Var Algorithmic changes
    - Weak constraint 4D-Var → sub-windows can run in parallel
  - Explore the use of EnKF
  - Reformulation of Non-Hydrostatic model

In Memory of Two benchmarkers



Bob Carruthers
CDC, Cray, SGI, IBM

Philippe Tesson
CDC, Cray, SGI

Questions ?