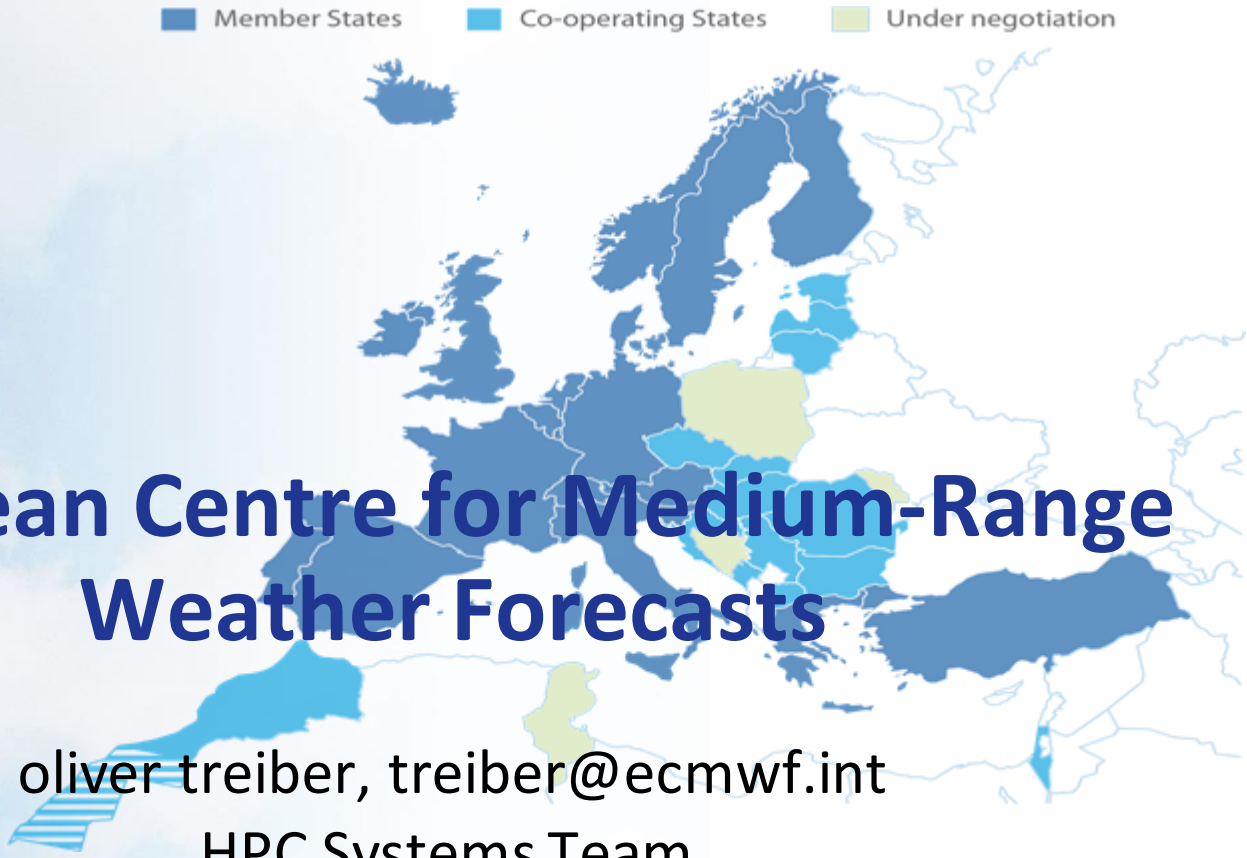


European Centre for Medium-Range Weather Forecasts

oliver treiber, treiber@ecmwf.int
HPC Systems Team




Overview

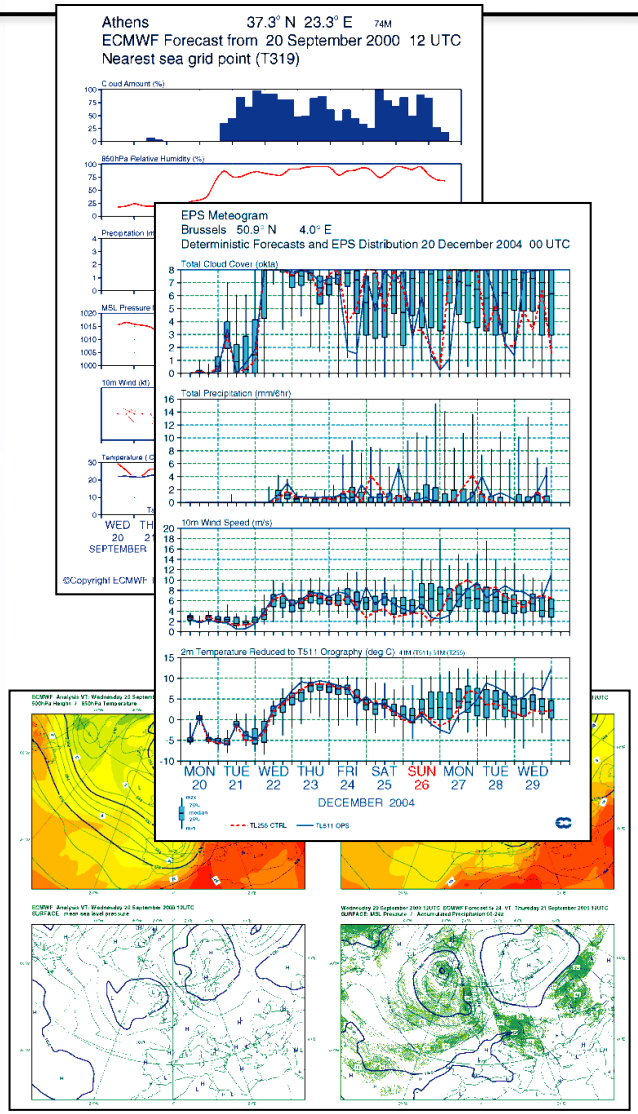
- what is ECMWF?
- HPC configuration (Cray and current production system)
- a hint at some ECMWF HPC idiosyncrasies

Who are we and what do we do?

European Centre — We are an independent international organisation funded by 34 States

Medium-Range — Up to fifteen days ahead. Today our products also include monthly and seasonal forecasts and we collect and store meteorological data.

Weather Forecasts — We produce global weather forecasts



What do we have to achieve this?

People — About 260 staff, specialists and contractors

Equipment — State-of-the-art supercomputers and data handling systems

Budget — £50 million per year

Experience — 37 years

A basic description of our models

Approximately 20 million observations

Data assimilation

First guess

Initial conditions

Model next step

Intermediate results

Archive

Prediction

The atmosphere is divided into about one billion cells each having associated parameters such as temperature, pressure, and wind direction.

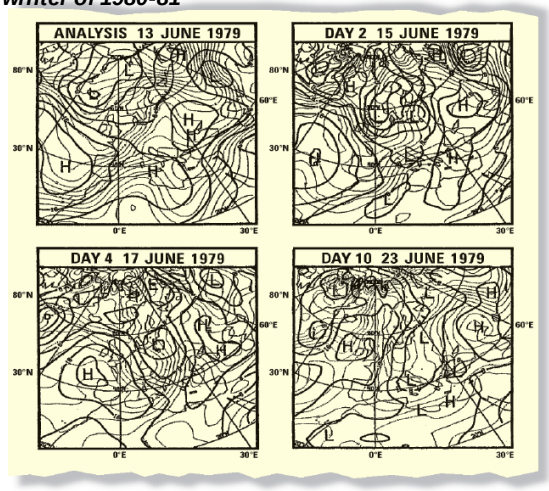
Laws of physics

The European Centre for Medium-Range Weather Forecasts (ECMWF) in 1979…

- On 1 August 1979,ECMWF delivered its first operational medium-range weather forecast using a Cray-1A.
- The model resolution was N48 (~200km) and used about 5 hours of CPU time to produce a 10 day forecast.
- Cray-1A
  - Single vector processor
  - 1 Mword (8 Mbytes) of memory
  - 12.5 nanosecond clock (0.08 GHz freq.)
  - 2 results per clock; 160 Mflops peak
  - ~ 50 Mflops sustained if memory sufficient
  - 2.5 GB storage
  - ~ 5 hours to produce a ten-day forecast, with overhead of 1 hour due to disk I/O

The Ambassador of Finland, in front of the Cray-1A, believed to be in winter of 1980-81

HRH Prince Charles speaking with Aksel Wiin-Nielsen and Rob Brinkhuysen during the opening ceremony of the ECMWF permanent headquarters at Shinfield Park in Reading in 1979



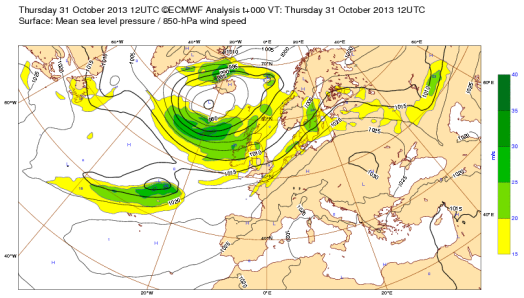
Forecast charts from 1979

ECMWF in 2014…

- ECMWF is an acknowledged world-leader in global medium-range numerical weather prediction.
- Meteorology and weather forecasts, including those of severe weather, are becoming increasingly important in an ever more complex world.

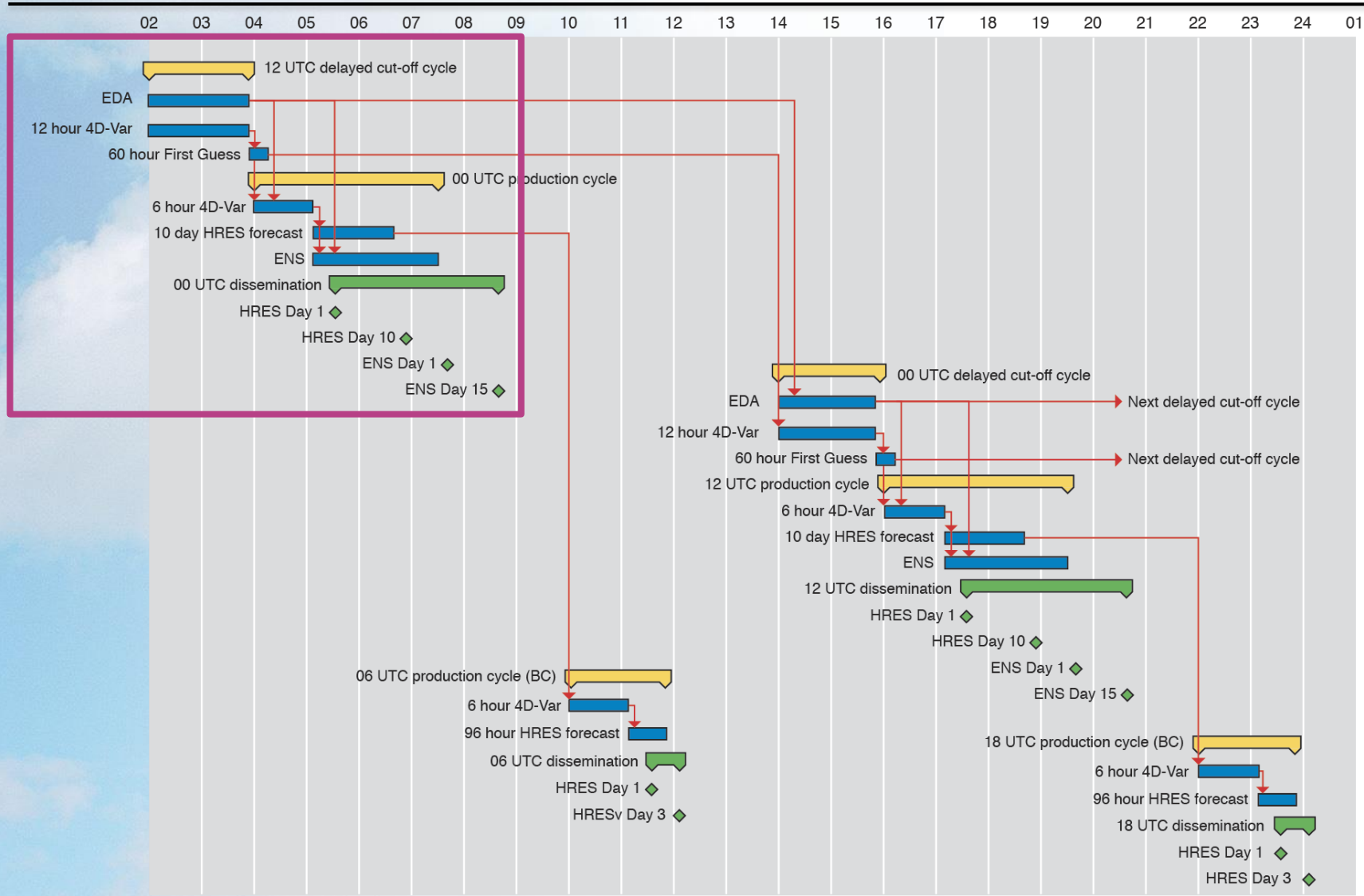
ECMWF permanent headquarters at Shinfield Park in Reading, UK

- High resolution deterministic forecast: twice per day
  16 km 137-level, to 10 days ahead
- Ensemble forecast (EPS): twice daily
  51 members, 30/60 km 62-level, to 15 days ahead
- Seasonal forecast: once a month (coupled to ocean model)
  41 members, 125 km 62 levels, to 7 months ahead
- IFS = ECMWF's Integrated Forecasting System; >70% of cycles
  - hybrid, >1M lines of source



ECMWF Forecast from 31 October 2013

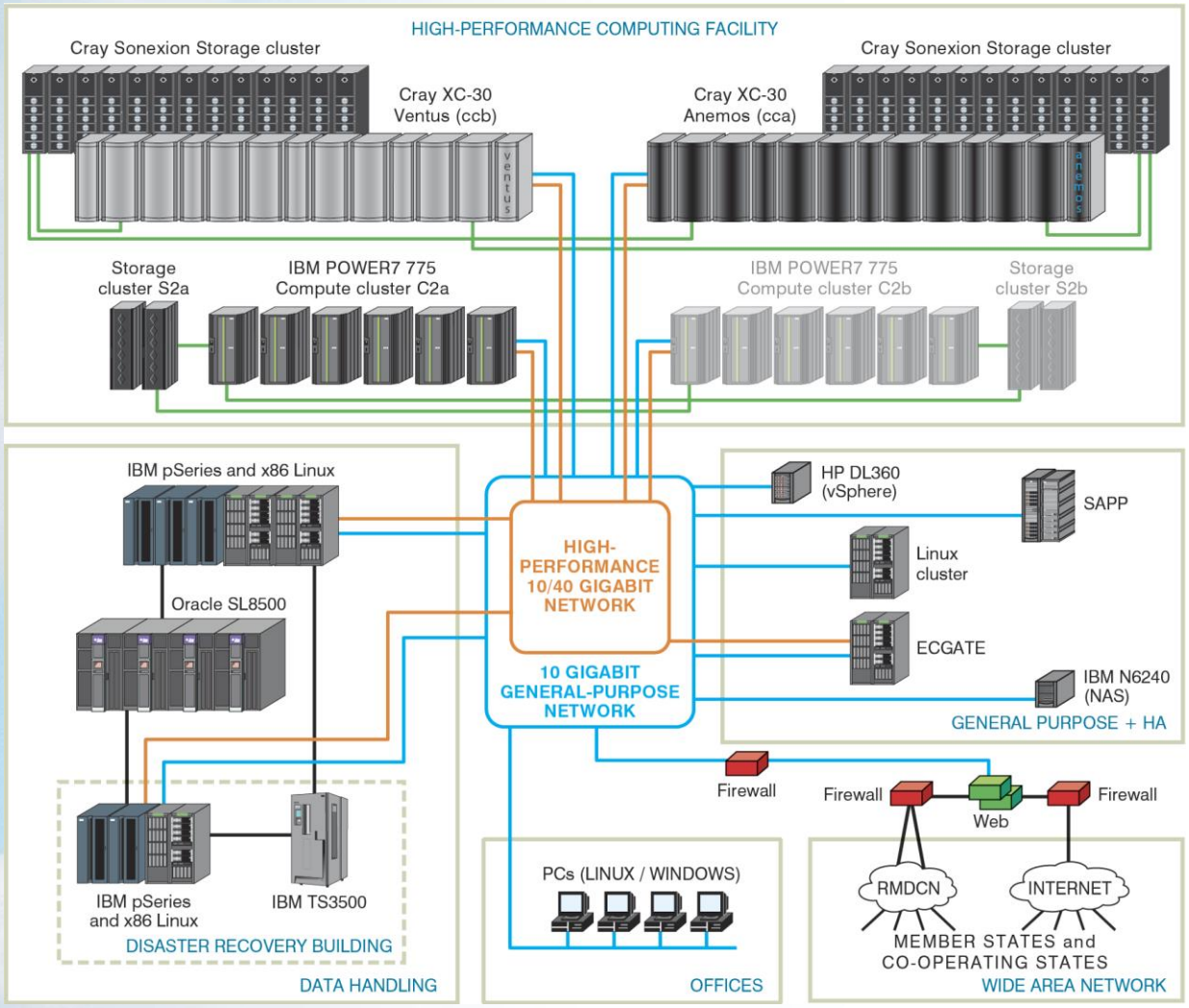
ECMWF x365 daily production workflow: not much room for downtime



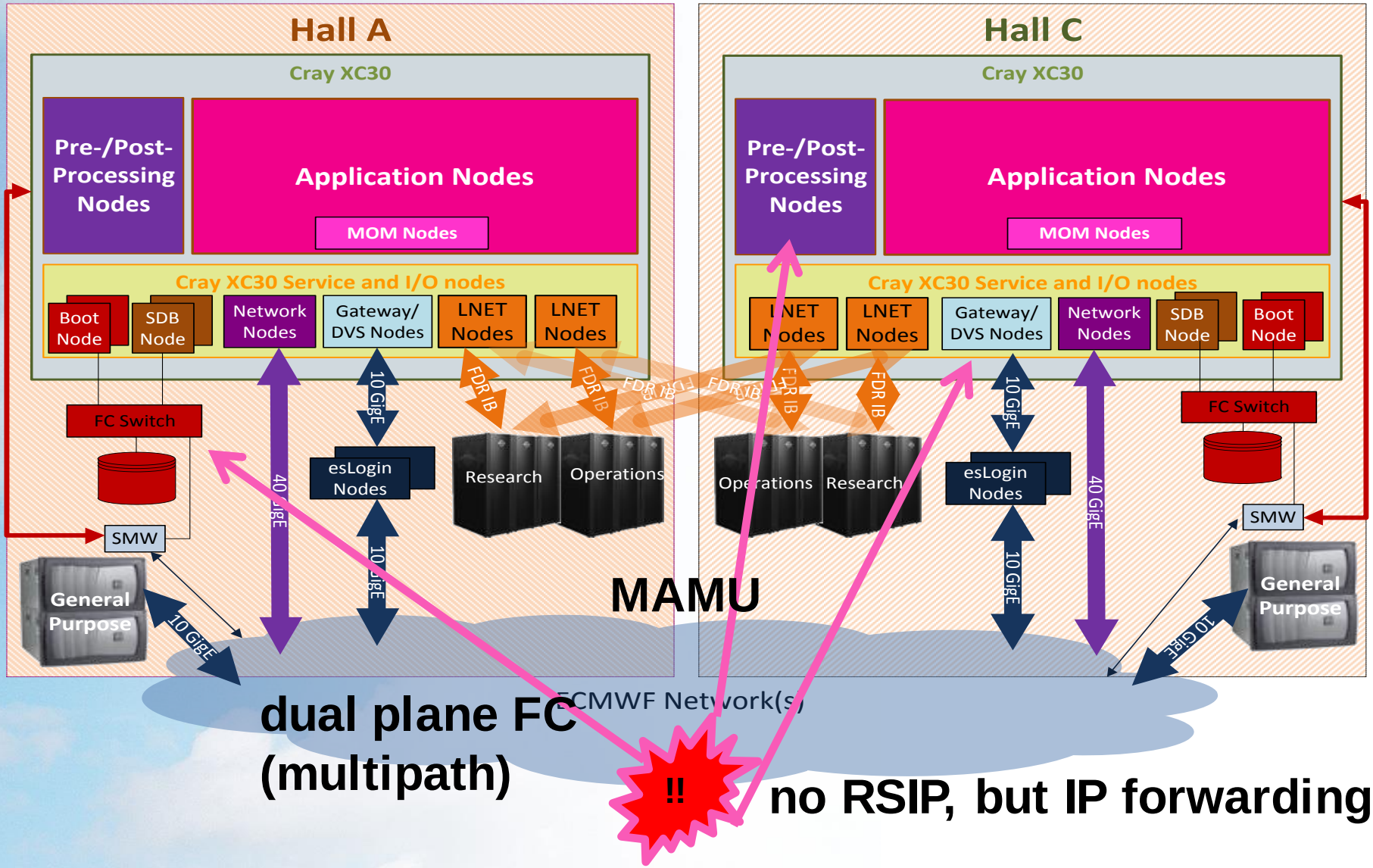
HPC at ECMWF: “two cluster set-up” (not good for top500…)

- since 2003: two “independent” compute clusters
  - but since ~2005: (MC-GPFS) with access to both storage clusters at same performance
- Idea is to have clusters at least as “self-sufficient”
  - could disable links to storage in alternate hall in case of major instabilities
  - a research and an operational filesystem in each hall, sufficient to run Ops and keep compute busy with research
  - backed by independent power and cooling
- Pros of shared storage - flexibility
  - workload can be flexibly distributed over the two compute clusters
  - operational suite can be moved from one compute cluster to the other by config variable (or spread over both), re-run failed jobs and move on, as long as currently used operational storage pool remains healthy and accessible (if not, restart using alternate filesystem from last checkpoints)
- Cons of shared storage – loss of complete independence as with air gap
  - hangs might spread over both compute clusters (e.g., “waiters” for GPFS); lustre recoverability might depend on other cluster

ECMWF System Overview

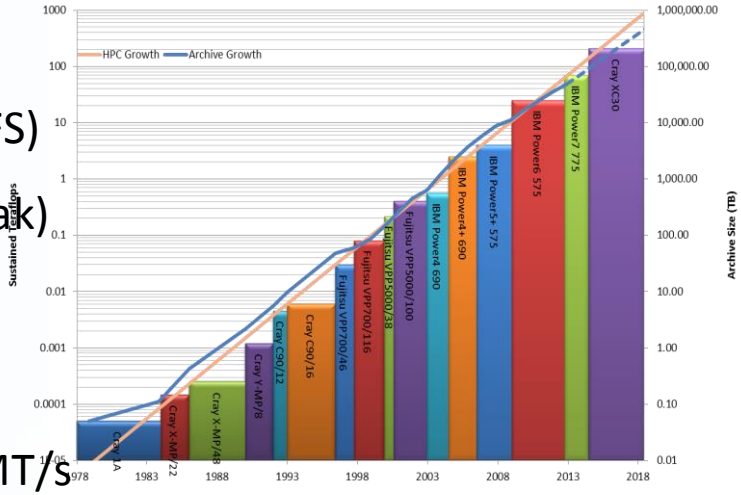


Overview of Cray system – currently being commissioned



2 Cray XC30 - Compute

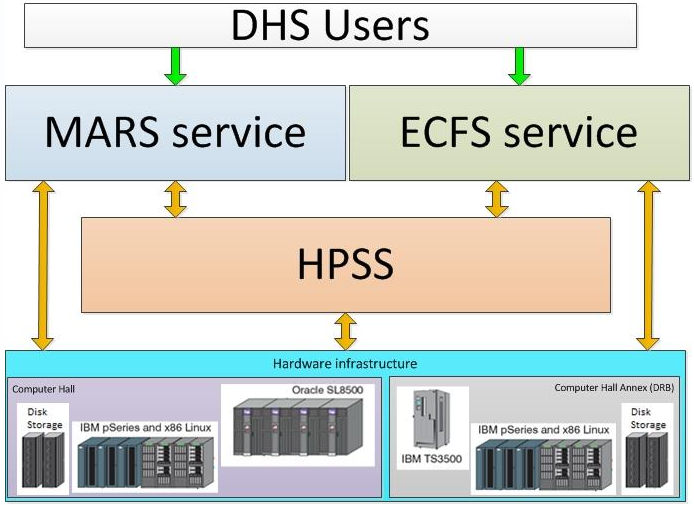
- 2 Cray XC30 systems in separate halls
  - self sufficient (but usually cross-mounted Lustre/NFS)
  - 210 TF sustained perf on ECMWF codes (~3.5TF/peak)
- each XC30
  - 19 compute cabinets
  - ~3,450 compute nodes (24c IvB), 64 GiB@1866MT/s
  - 60 pre/post processing nodes (24c IvB), 128GiB@1866MT/s
  - 4 pre/post p nodes/services, 256GiB@1333MT/s
  - 7 aprun-MoMs, LNET, 8 DSL, 2 DVS,
  - 8 data transfer nodes (40Gb bonds to LAN), in lieu of RSIP (as that has currently no HA)
- Compared to Cray-1A:
  - ~21M times faster, ~60M times main memory capacity



the Cray installation team

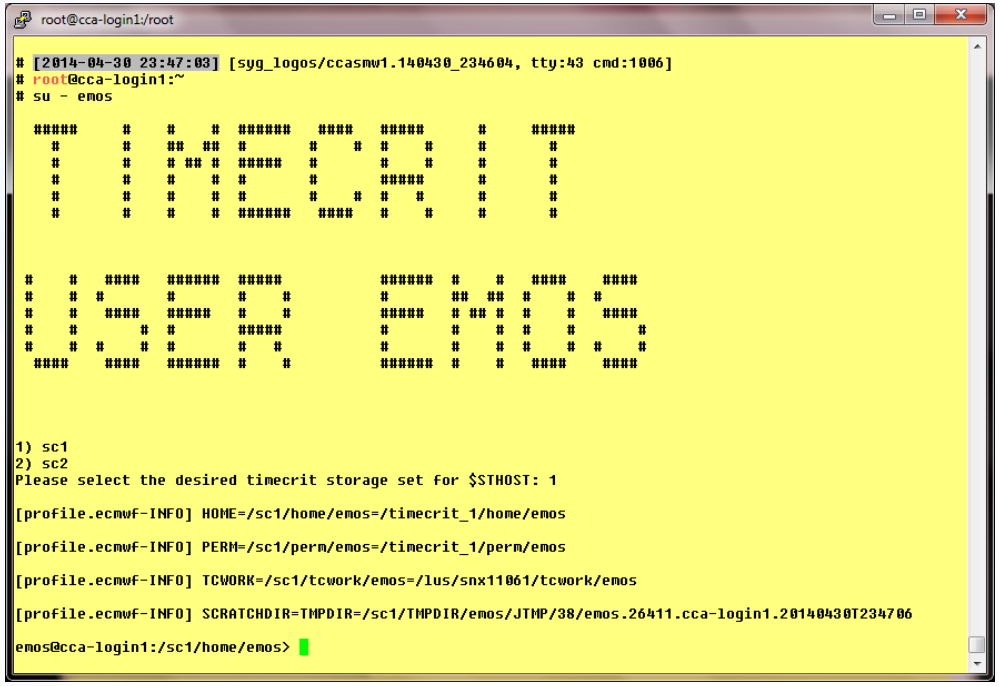
Cray XC30 - Storage

- 2 Sonexion 1600 in each hall
  - one FS for timecrit activities; one FS for research projects; identical config in other hall (no system-provided replication across the halls)
  - cross-connected to XC30 LNET in alternate hall via IB (100m fibre), same access bandwidth
  - 400GB/s & 12 PB aggregate (evenly split across halls)
- each hall: NetApp FAS6240, 38TB net, for NFS
  - $HOME, project HOMEs, /usr/local
  - async snapshot replication to stand-by LUNs across halls
- 100+ TB/d new data archived to HPSS
- Compared to Cray-1A:
  - ~4.5 million times the disk capacity
  - ~100k times streaming bandwidth (!)



Storage sets for time-critical activities

- each hall has one operational lustre/NFS filesystem set, identical config
  - accessible from both compute clusters
  - no system-provided replication on lustre (some weather sites did that with GPFS)
  - environment variable for batch job selects storage set for $HOME,$TMPDIR; selection dialog at interactive session startup; no scp to "$HOME"
  - otherwise, depend on /usr/local/ (separate FS via NFS per compute cluster) only

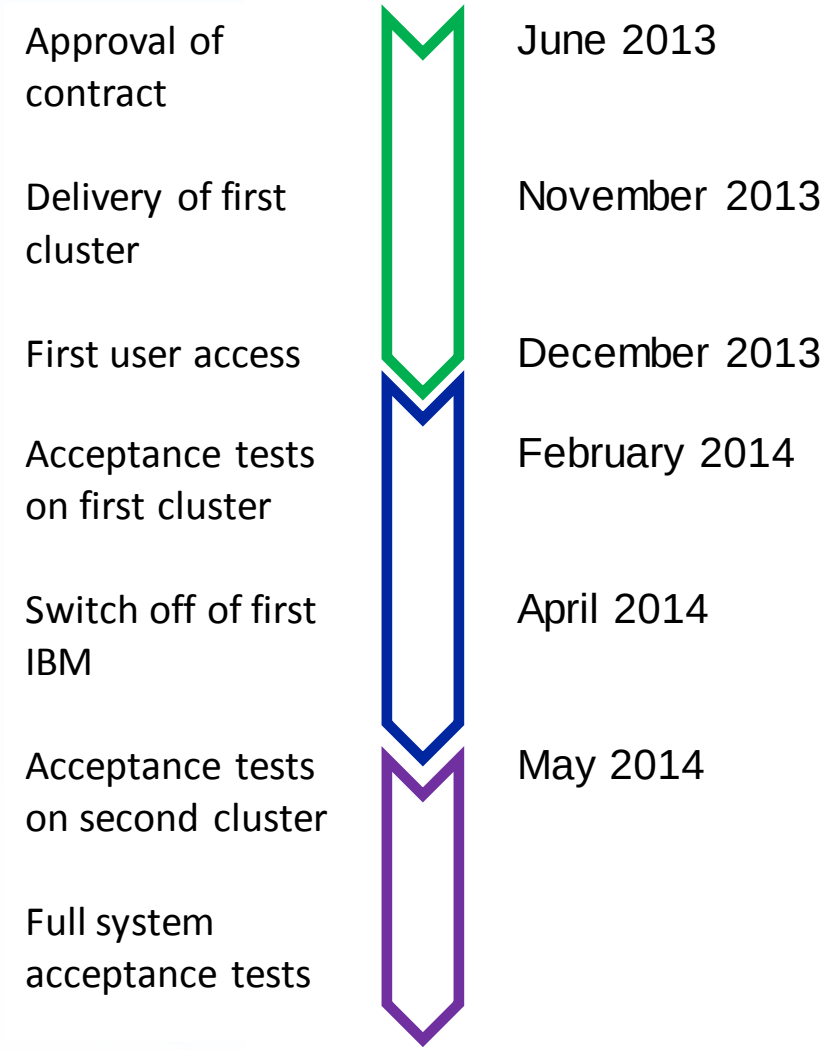


pre/post-processing (PPP) on repurposed XC30 nodes (MAMU)

- **PPP nodes run several concurrent jobs, serial and small parallel**
  - Cray provided new mpiexec for intra-node (sharedmem) MPI in lieu of aprun
    - Cray MPI library supports both aprun/ESM and mpiexec/MAMU mode: MAMU=multi app,multi user, per node
- PPP nodes run as normal linux server cluster nodes, each with its MoM, technically they are service nodes (no ESM, snvar via NFS)
- Jobs share PPP nodes (as is case on aprun-MoMs, too),
  - we required water-tight enforcement of memory limits per job, not just periodic polling of process table
    - on PPP, to be implemented via hook that runs jobs in separate cgroups
    - plan to use on aprun MoMs for safety, too;
- if necessary, could repurpose more (or less) 64GB ESM nodes as PPP nodes; as of 5.2, only requires reboot of nodes to be repurposed

System Acceptance Status

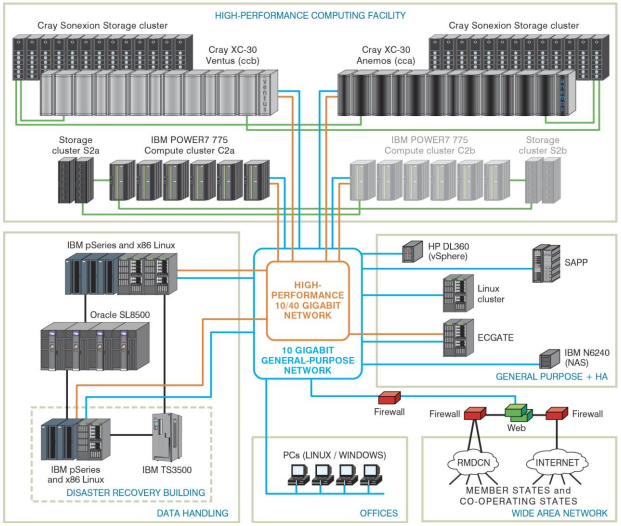
- Acceptance in stages
  - EC requires to always have two clusters able to run operational forecasts
  - Power and cooling restrictions allow only three 1-1.5MW clusters at once
- Acceptance process
  - Stages (=kit in one hall) and entire System (=all kit) undergoes:
    - 5 day Functional Verification Test
      - failover (cable pull, node shooting), benchmarks,…
    - +30 day Operational Test
      - tracking Downtime, Incidents & Events targets
    - +60 day further Reliability Test (for Phase only)



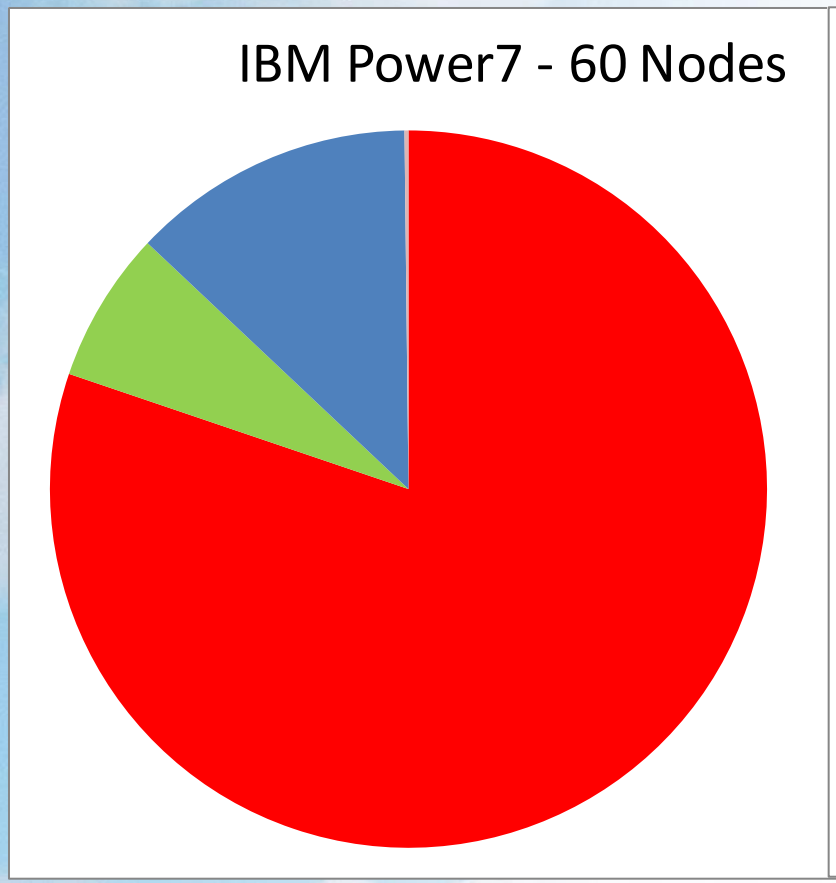
Current system

- **IBM POWER7 System, Torrent/HFI interconnect**
  - Two identical systems for resiliency
  - 70 Teraflops sustained on ECMWF's codes
  - Each cluster
    - 754 Teraflops peak performance
    - 23,648 processor cores (739 compute nodes)
    - 1.5 Petabytes storage
    - Power consumption: ~1.2 MW

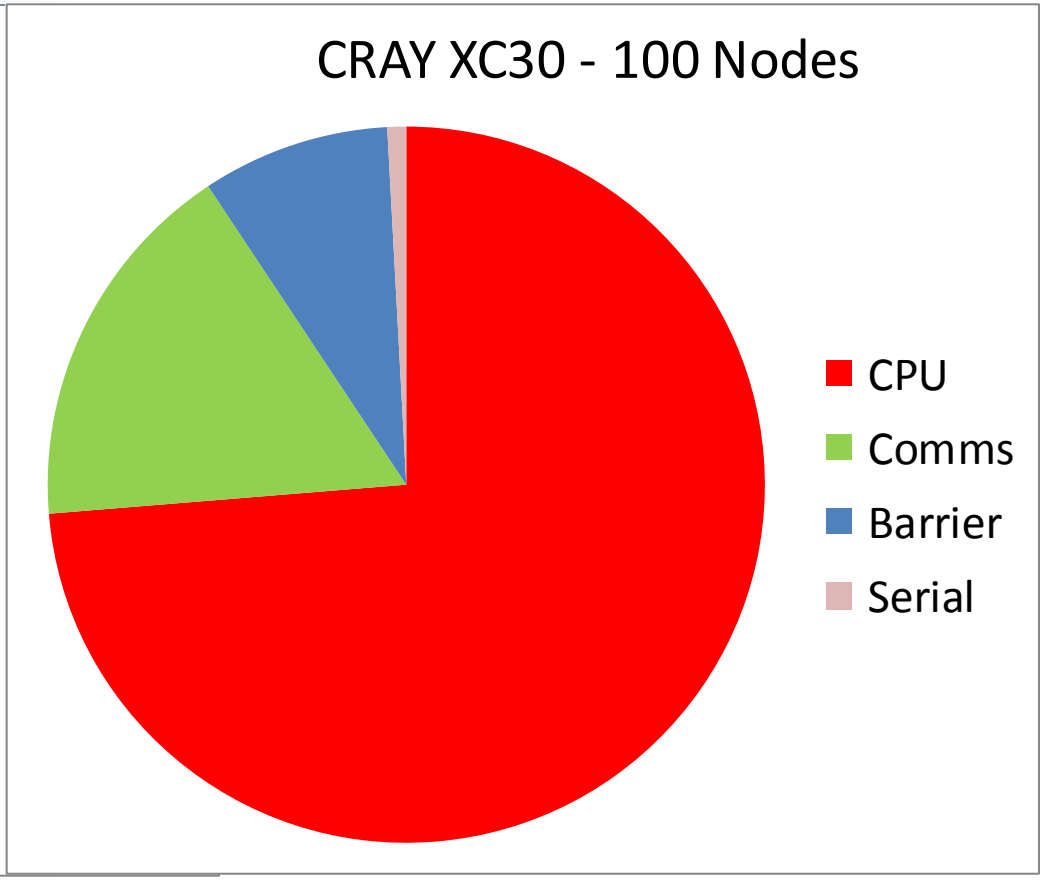
- **Allocation of ECMWF's computing resources**
  - 25% Operational activities
  - 50% ECMWF research activities
  - 25% Member State users (throughout Europe)



Performance: T1279L137 10d HiRes FC, current op. resolution (without output to disk)



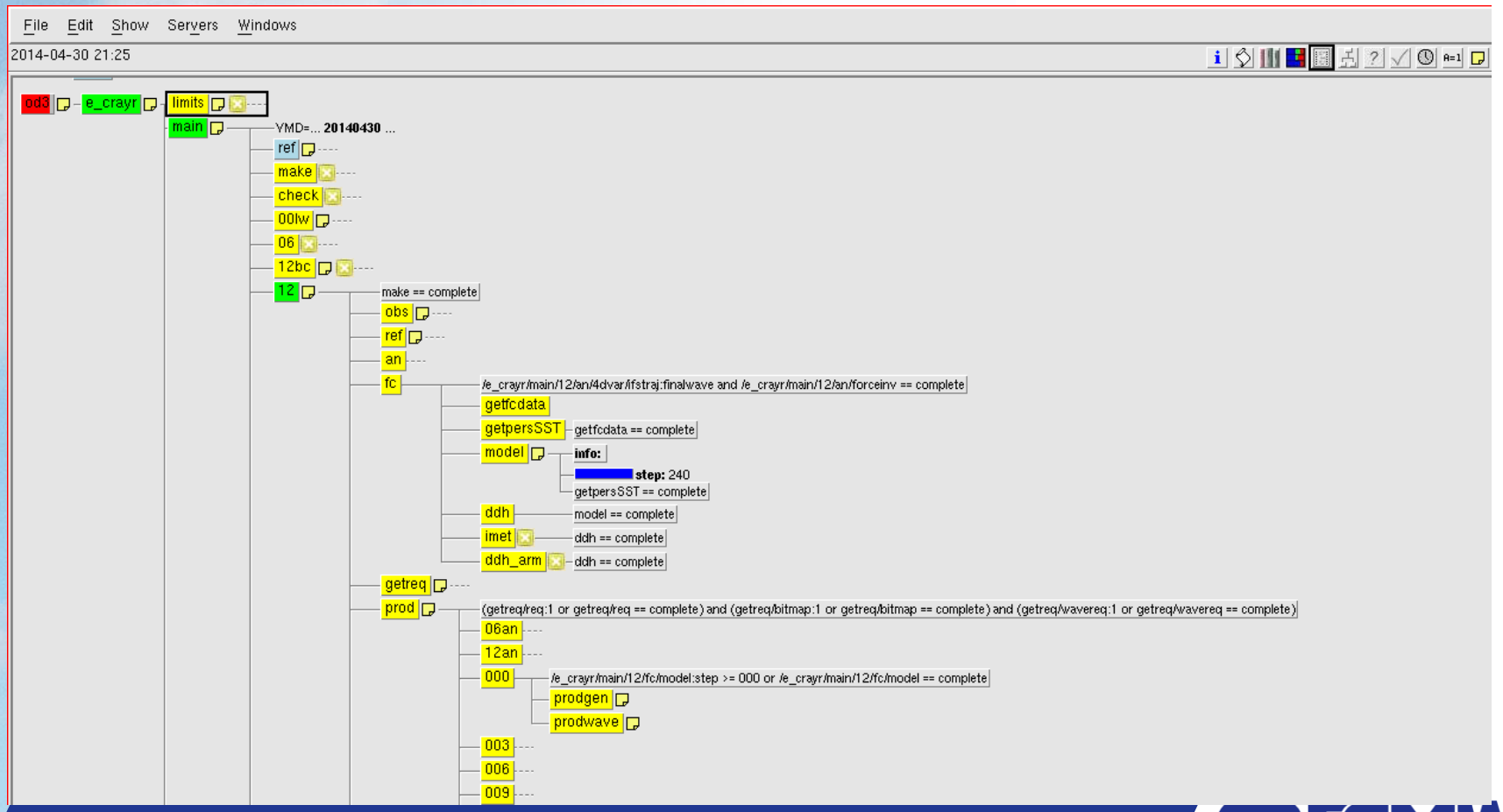
2258 seconds
5.1 Tflops (8.6% peak)



2182 seconds
5.2 Tflops (10.4% peak)

Workflow: ECMWF Supervisor Monitor Scheduler (Xcdp GUI)

- organises workflow as dependencies between tasks submitted to HPC
- ECMWF researchers usually do not directly interface with batch system
- flex_submit scheme inspects HPC queueing times and balances over clusters



ECMWF scheduling idiosyncrasies

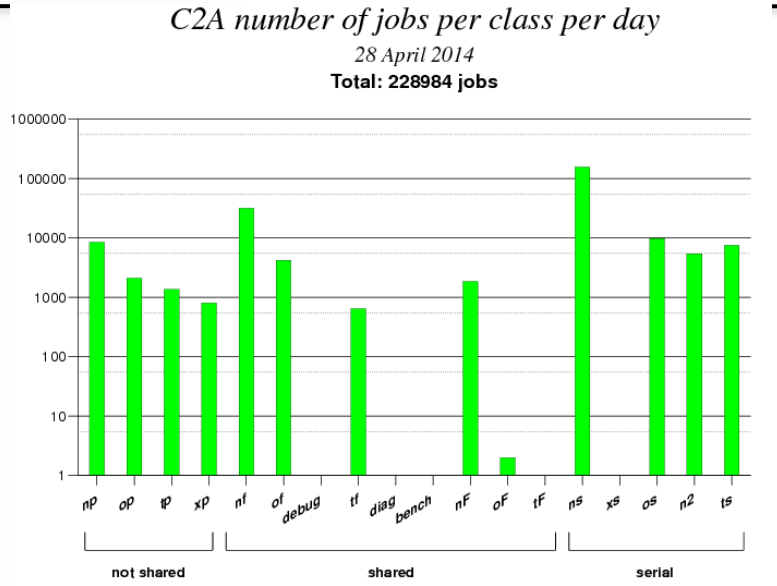
- no cluster dedicated to operations, but usually have a “sticky” choice for primary operational cluster
- operational workload must fit into one cluster; has variable footprint
- no preemption/suspend/resume scheme used
  - but want to fill up cluster with research to max utilisation
  - flat ESM domain, not yet tried topology aware scheduling (too expensive?)
  - flexible reservation schemes using variable “number of nodes-for-research-licences” to guarantee sufficient free resources for operational needs; pre-idling with advance reservations invisible to operational workload (only blocking/visible to research workload), so binding of operational jobs to ARs
  - wall-clock prediction based on lookup from runtime-history DB by external submit filter
    - add +24hrs walltime offset for research workload, +6 hours for operations (also a “grace” period, but underlies the AR use scheme)
  - this has been in use for last 10y on LoadL, currently adapting to PBSpro

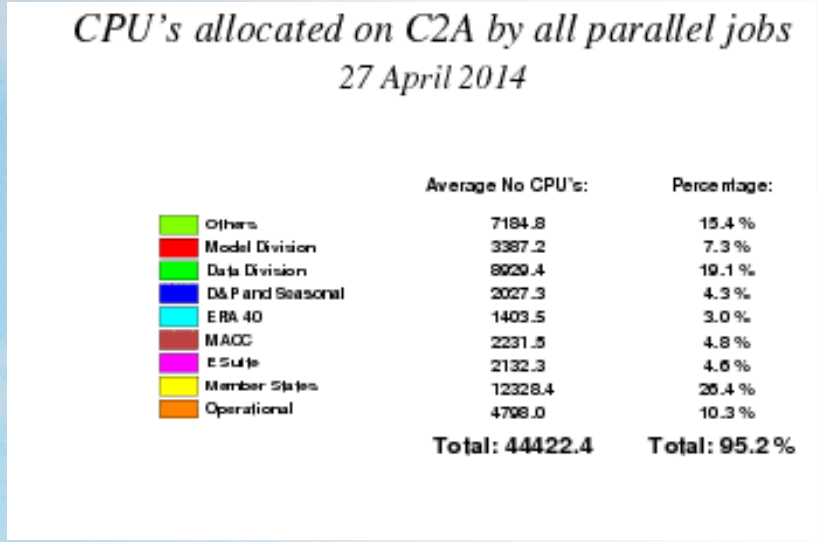
ec_jobs = operators' view on batch queue on current system

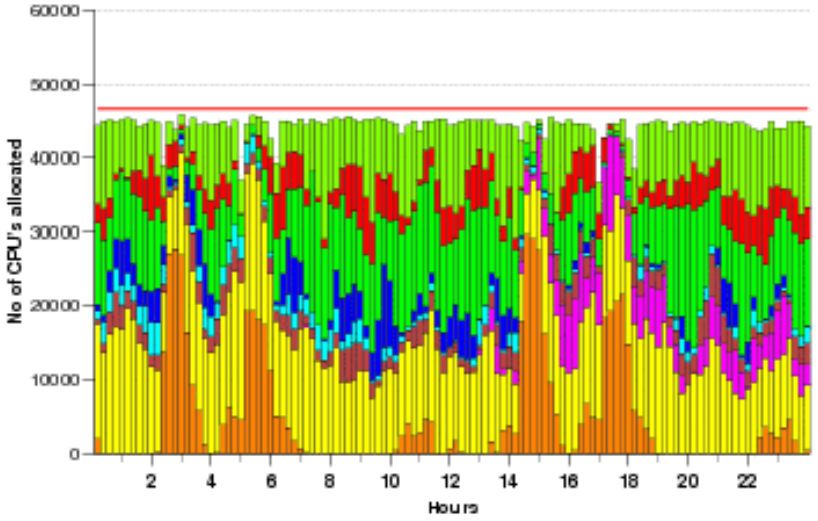
```
21:56:00 avlop:24 DRN:2  wait: np:143:24j-254m ns:90:2j-0m tp:0:1j-0m
  at 21:55:14: c2a1u.2276.r/emos18bcF in 58min has 72/72
  at 21:55:14: c2a1u.2275.r/emos18bcK in 23min has 72/72
  at 21:55:14: c2a1u.2274.r/emos18bcA in 14min has 36/36
be be be de di ep es n2 nF np ns oF of op os tF tf tp ts      W.Clock     Prio   u-10/s10|u-2m|utot|MBnow|MBhwm|IOlowbn Finish  Nodes
tow      arpegeV6-nemo1          1  c2a9u.15141716.1    np  00:36 24000-50  31.6/0.4|31.5|31.7|39592|39592|  28897 -00:00  827
en2      OBS_INC_PREV_FC_LETKF   2  c2a9u.15142621.0    np  00:00 24000-50                                 N/A -00:00  254 434
rdx-pat  g3lu_legA_modeleps_nemo 2  c2a3u.21015141.0    np  00:18 23700-50  22.7/0.0|22.5|22.2|18694|18880|   2461  00:00  731 753
rdx-pat  g3lu_legA_modeleps_nemo 2  c2a3u.21015147.0    np  00:18 23700-50  22.4/0.0|22.4|22.2|18694|19018|   2461  00:00  755 756
rdx-pat  g3lu_legA_modeleps_nemo 2  c2a7u.14955698.0    np  00:19 23700-50  22.4/0.0|22.2|21.9|18563|18563|   2369  00:00  757 760
eras     1889_ifstraj_uptraj_0   4  c2a9u.15142584.0    np  00:02 24000-50                                 N/A  00:00  116 127 1a1 880
rdx-dea  g39l_surf_anal_ph2ana   1  c2a3u.21019837.0    nF  00:00 18700-50                                 N/A  00:00  3b3
tow      arpegeV6-nemo1          1  c2a9u.15141672.1    np  00:39 24000-50  32.0/0.0|31.6|31.7|39589|39589|  54491  00:01  822
rdx-pat  g3lu_legA_modeleps_nemo 2  c2a7u.14955720.0    np  00:18 23700-50  21.7/0.1|22.3|22.2|18696|19023|   2368  00:01  286 752
rdx-pat  g3lu_legA_modeleps_nemo 2  c2a3u.21015170.0    np  00:18 23700-50  21.8/0.1|22.2|22.2|18485|18650|   2368  00:01  4c5 557
ocx-ne1  g359_outer_an0          16 c2a1u.22674129.0    np  00:06 23700-50  26.8/0.2|26.8|22.4|19104|19104|6365233  00:01  120 240 252 263 287 2b1 416
tow      arpegeV6-nemo1          1  c2a9u.15141755.1    np  00:37 24000-50  31.9/0.0|31.6|31.7|39629|39629|  28739  00:01  490
tow      arpegeV6-nemo1          1  c2a7u.14959517.1    np  00:37 24000-50  31.1/0.1|31.6|31.7|39636|39636|  28896  00:01  231
rdx-stmk g3hp_uptraj_0_ifstraj   30 c2a4u.21603450.0    np  00:07 23700-50  25.2/0.2|11.9| 9.1|33653|33653|  97181  00:01  314 324 331 336 340 342 346
                                                                                                                        520 646 716 732 766 816 926
rdx-std  g3q2_uptraj_0_ifsmin    6  c2a4u.21610334.0    np  00:12 23700-70  25.4/0.2|25.2|24.0|20927|20927|   7943  00:02  272 321 337 362 3a4 3a6
fa1o     c2a9u.15142517          1  c2a9u.15142517.0    np  00:07 24000-50   3.2/0.1| 3.4| 3.0|29264|29264|   8513  00:02  974
smk      HIRLAM e4m2_ecgb Date Hou 3 c2a3u.21019753.0    np  00:06 24000-50  31.0/0.0| n/a|30.3| 7183| 7183|   1831  00:02  654 982 984
rdx-sts  g3i3_surf_anal_snow     1  c2a9u.15141567.0    np  00:03 23700-75                                  N/A  00:02  2a7
tow      arpegeV6-nemo1          1  c2a7u.14959497.1    np  00:39 24000-50  32.0/0.0|31.6|31.6|38999|39285|  53653  00:03  4c4
emos     sekf_surf_anal          60 c2a9u.15142301.0    np  00:09 24800-70  27.9/0.1|25.8|25.0|28752|28752|  13778  00:04  112 122 124 126 135 196 226
                                                                                                                        4b6 516 521 541 5a3 667 673
                                                                                                                        857 860 883 912 931 933 935
rdx-std  g3bs_uptraj_0_ifstraj   6  c2a7u.14959330.0    np  00:05 23700-75                                  N/A  00:04  281 284 291 2a2 2a4 666
ocx-dihz g2tk_outer_an0          8  c2a4u.21610526.0    np  00:08 23700-50  27.0/0.1|20.7|20.0|22785|22785|3098092  00:04  583 584 5a1 5a2 5b0 5b4 5b6
rdx-std  g2lx_uptraj_0_ifstraj   6  c2a3u.21018797.0    np  00:05 23700-75                                  N/A  00:05  2b7 375 425 457 473 477
eras     1889_model_fclong       4  c2a3u.21019790.0    np  00:07 24000-50   9.2/0.1| n/a| 8.0| 5874| 5814|   702  00:05  430 441 446 721
rdx-std  g2nb_uptraj_0_ifstraj   6  c2a1u.22673214.0    np  00:05 23700-75                                  N/A  00:05  517 536 547 551 572 573
nlk      c2a1u.22673837          2  c2a1u.22673837.0    np  00:23 24000-50  32.0/0.0|31.9|31.7|37962|37962|  20730  00:06  930 967
rdx-pat  g3lu_legA_modeleps_nemo 2  c2a3u.21015480.0    np  00:18 23700-50  22.5/0.0|22.4|22.0|18686|18920|   1541  00:08  140 151
eras     1906_monthmean162_stat  2  c2a3u.21019812.0    np  00:10 24000-50                                  N/A  00:09  125 193
eras     1906_monthmean162_stat  2  c2a7u.14960370.0    np  00:10 24000-50                                  N/A  00:09  947 971
rdx-nat  g3ll_fcgroup1_model     4  c2a9u.15137392.0    np  00:58 23700-50  30.1/0.0|29.1|29.5|28052|28497|  10174  00:10  773 840 844 871
emos     ifsmin_uptraj_2         60 c2a6u.22126760.0    np  00:10 24800-96                                  N/A  00:10  115 121 144 150 162 166 191
                                                                                                                        414 420 476 496 4a4 4a6 4a7
                                                                                                                        682 692 697 6a1 6a3 6a4 6a5
rdx-sts  g3i3_uptraj_2_ifsmin    30 c2a9u.15142382.0    np  00:18 23700-74  27.5/0.1|26.8|24.2|44670|44670|  68116  00:10  123 160 232 2c3 310 355 364
                                                                                                                        640 647 650 662 663 672 694
rdx-nesk g35r_uptraj_2_ifsmin    30 c2a3u.21019626.0    np  00:18 23700-74  28.2/0.1|26.8|24.1|44503|44503|  59446  00:11  154 156 213 225 261 264 273
                                                                                                                        761 843 864 872 913 917 923
eras     1906_cpmodel_nemo6_subchu 5 c2a4u.21609319.0    np  01:02 24000-50   6.8/0.0| 6.8| 6.8| 7144| 7182|  40930  00:12  1a3 595 5a5 750 851
nlk      c2a1u.22674028          2  c2a1u.22674028.0    np  00:23 24000-50  32.0/0.0|31.9|30.8|37847|37847|  11986  00:15  146 164
nlk      c2a4u.21610430          2  c2a4u.21610430.0    np  00:23 24000-50  31.9/0.0|31.9|31.5|37920|37920|  17340  00:15  175 1b0
rdx-ste  g2fq_uptraj_2_ifsmin    30 c2a1u.22674138.0    np  00:17 23700-74                                  N/A  00:16  110 114 131 133 136 174 1b5
                                                                                                                        811 817 874 882 922 932 937
nlk      c2a9u.15142486          2  c2a9u.15142486.0    np  00:23 24000-50  31.9/0.0|31.9|30.9|38057|38057|  16793  00:17  6b4 725
frjr     mi-rerun4.glu_var_anal_n1 2 c2a6u.22126651.0    np  00:26 24000-50  26.9/0.0|26.9|25.7|18421|18421|  13353  00:20  764 765
<more>
...
```

cpu allocation on current operational cluster (p7iH)

- **usually >95% allocation operational cluster, even with adv reservations for timecrit workload**



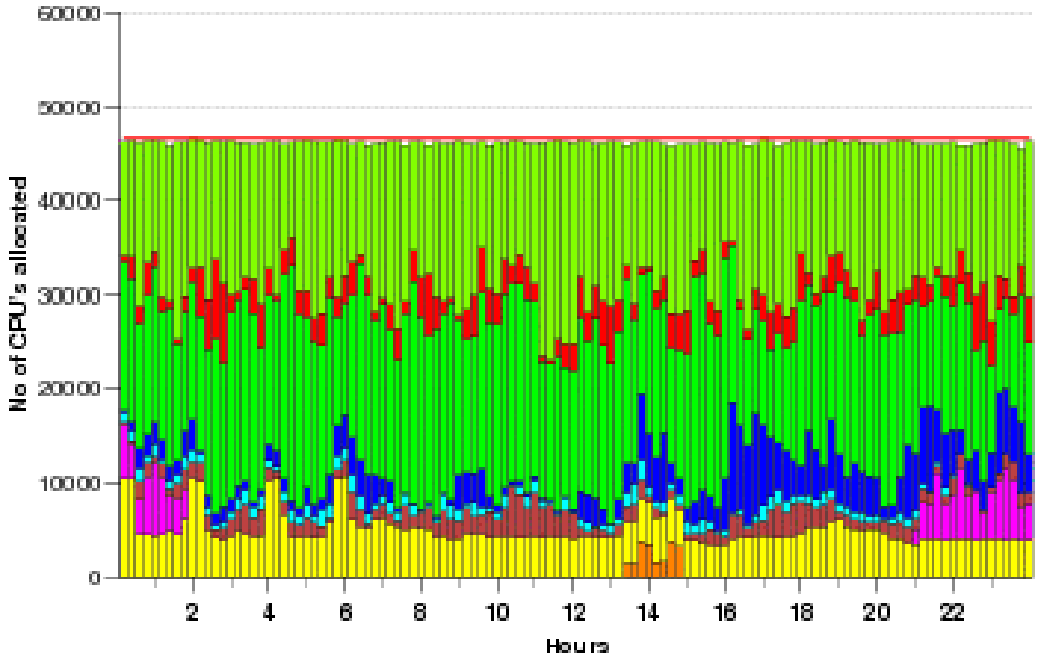




current production back-up cluster daily allocation, 175k j/d

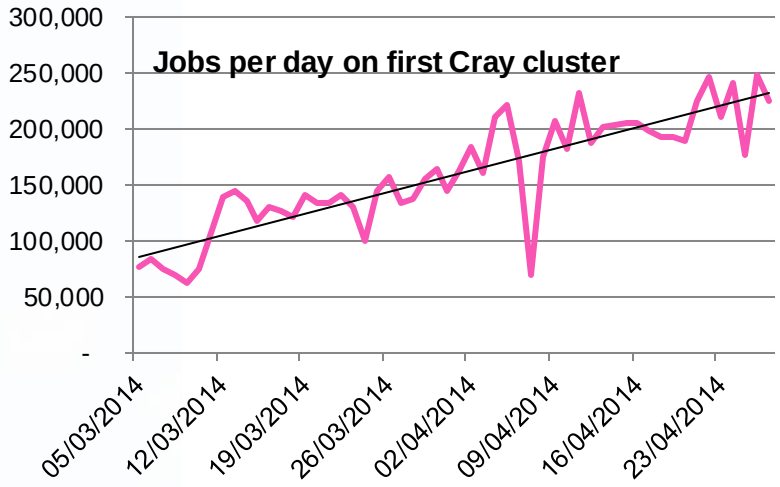
CPU's allocated on C2B by all parallel jobs
06 April 2014

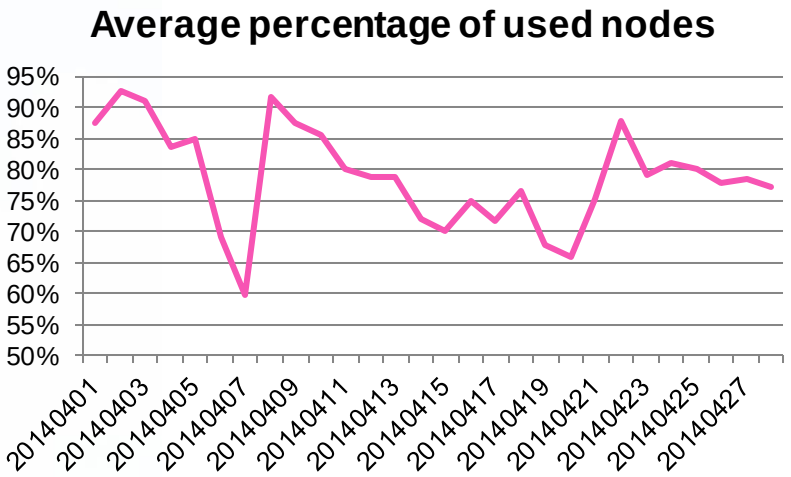
	Average No CPU's:	Percentage:
Others	15458.6	33.1 %
Model Division	2761.0	5.9 %
Data Division	16224.4	34.8 %
D&P and Seasonal	3169.5	6.8 %
ERA 40	865.0	1.9 %
MACC	1678.0	3.6 %
ESuite	1006.3	2.2 %
Member States	4906.8	10.5 %
Operational	173.7	0.4 %
	Total: 46243.2	**Total: 99.1 %**



scheduling on Cray: PBSPro

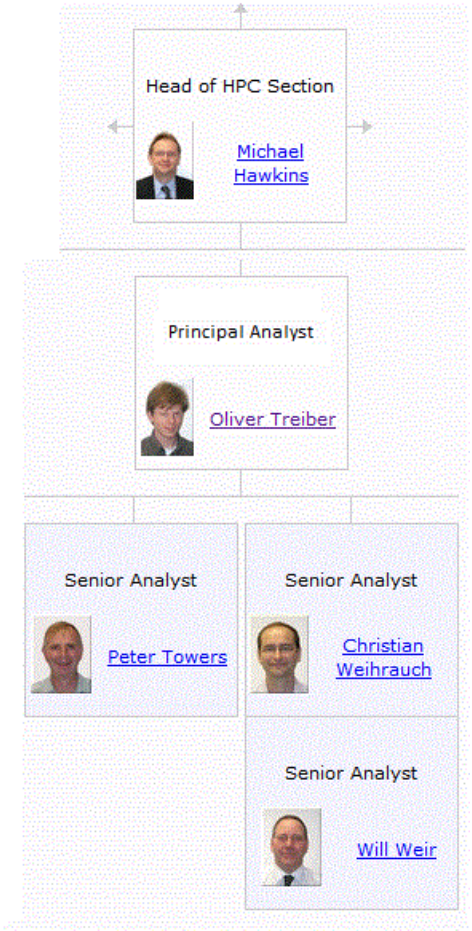
- A lot of throughput computing
  - biggest operational jobs will be concurrent 1x300nodes highres forecast alongside 50 ensemble forecasts of ~30nodes, for an hour
  - lots of concurrent parallel and serial postprocessing/product generation, etc
  - many concurrent research experiments, usually fewer nodes than operations
  - some bigger research jobs > 1000n exploring future resolution upgrades
  - lots of short jobs (<2 minutes)
  - currently in need of "scheduling" the scheduler (change queue state from cycle to cyle, etc)
- currently configured as one PBS complex per cluster; inter-job dependencies are managed through ECMWF SMS,
- exploring splitting into one parallel and several postproc complexes per cluster for scheduling scalability





Thanks for your attention!

ECMWF HPC team



Cray team at ECMWF:
Pete Custerson, John Hopewell (PrjMgr), Chris Spiller, Alex Wood