

White Paper: Managing and Increasing Fusion Computing Resources.

Ronald E. Waltz

Bruce I. Cohen

Dalton Schnack

Carl Sovinec

Steve Jardin

Don Batchelor

Presented to the Theory Co-ordinating Committee (TCC)

APS 2004

White Paper: Managing and Increasing Fusion Computing Resources.

There has been a growing disconnect between the OFES Fusion computing program goals and the availability of DoE computer resources. The present disparity between goals and resources within the base Fusion computing and theory program will become much worse if the recently proposed OFES/OASCR Fusion Simulation Project were to be funded without the proper mix of computing resources. The problem is not solely attributed to insufficient funding but arises in part from procurement of high cost capability computing (cycles/sec –single supercomputer) without a proper mix of low cost capacity computing (cycles/year – many cluster-computers). In fact at present there is almost no mixture at all.

The problem:

Nearly all large scale Fusion computing is done on the 6656proc (processor) Seaborg supercomputer at NERSC (LBNL) with some smaller supercomputers coming online in the last year at the CCS (ORNL). Most Fusion users are at NERSC. Fusion is the dominant user: 32% in FY03 with the next (Chemistry) at 16%. 42% of Seaborg usage is on less than 256proc =16nodes which could easily be done on smaller “clusters”. 42% is used in the 512-2048proc=32-128node range, with the remainder intermediate and some larger than 128nodes. Supercomputers are only cost effective for jobs which really need a large number of processors. OMB insists that 50% of Seaborg jobs be over 512proc, i.e. about 1/12 of Seaborg. But here is the weakness of time sharing on a large supercomputer: For a perfectly scaled code, a job should run 2x faster on 2048proc than on 1024proc. We appreciate that some codes are limited by memory per processor and require more processors. However if all users have the same priority, it stands to reason that a user will likely have to wait longer to get his/her turn at 1/3 of the computer compared to 1/6. It may well be that turn-around time for the job is actually quicker on 1024proc than 2048proc. We will call this “inversion”. It is the turn-around speed that counts for scientific productivity - not the speed of 6656 processors on a single problem - the advertised 10 teraflop speed. Supercomputers make sense only in an environment of dedicated use by a necessarily limited number of users. Open time sharing is usually inefficient when a single supercomputer must handle a broad mix of small and large jobs. To encourage large jobs justifying supercomputers and prevent inversion, large jobs are given a priority or discounted charges, which squeezes the small jobs with nowhere else to run. Few codes have perfect scaling at fixed problem size, so this leads to inefficient use of cycles.

The usual management goal (per the NSF SDSC) for timesharing on a parallel process supercomputer is to obtain better than 50% loading efficiency (E): $E = \text{run_time} / \text{turn_around_time} = (R) / (W+R)$ with $R = \text{run_time}$ and $W = \text{wait_time}$. Ideally E should be better than 50% across all processor count. Seaborg has become so overloaded that E has dropped well below this. The **Appendix** gives **tables** of NERSC statistics on E and W during the four quarters of 2004 (Regular Class excluding INCITE jobs). The overall job average E was 20.1% but dropped well below 10% in the fourth quarter. E=20.1% is due in part to the surprising fact that most of the jobs have very short

run times R (Reg. Class: 58% run for less than 1/2 hour and 41% run for less than 5 minutes). It is unclear if these are test jobs or failed jobs. Should average E be weighted by jobs or time spent on the computer? For jobs run over 8 hours, the job average was $E=30.1\%$. We expect E for a job to consistently increase the longer it stays on the computer. Indeed jobs approaching the 48hr limit (allowed for jobs larger than 512proc) get to $E=40\%$ (see **last table**). Ideally, E would be nearly constant for small and large node count jobs. There has been much complaint about long waits for small node count use (the policy disfavoring small node count jobs has vacillated). The **tables** appear to show that efficiency for small node count is not necessarily worst than large node count. They do show some examples close to “inversion” i.e. turn-around time breakeven where E is half that for half the number of processors.

The miserable Fourth Quarter loading efficiency is for users with “no priority boost”. In mid-summer NERSC announced (under pressure from OMB) that jobs larger than 512proc would get a 50% discount on the charge time. This greatly over allocated the computer. In August Fusion computing came to a near stand still because no Fusion user had a priority “boost” of any significance and 3 INCITE PI’s (with 10% Seaborg annual allocation) decided to use most of their time within 1-2 months. They were given a “head-of-the-line” boost. (For INCITE users, E could approach 100%.) Combined with a large (and discounted) allocation, they shut out nearly all other use while competing only against themselves. This extreme boost was completely unneeded. A simple “2-3 day age boost” would have given the INCITE work adequate priority. This was a management induced problem! The problem has abated recently as INCITE has finished.

Loading efficiency E reflects an average turn-around speed, but does not reflect the full loss of human productivity. Many of us have had the experience of waiting 3 weeks for a 48hr 512ps job ($E=8\%$) only to find it bombed out on errors in the input file ($E=0\%$). One cannot over emphasize the importance of minimal waits in code development and cutting edge research where the codes are just barely working. Usually one must look to see the results of one run before putting the next into the batch queue. If a jobs starts quickly and can be monitored, one can terminate the job if it is not going well and restart with modified sources or new inputs. We appreciate that the batch wait environment is not a problem for all users. For example many chemistry users are likely using mature production codes which can stack up many batch runs. This is simply not the way most large scale Fusion computing is done. It should be noted that previous generation (e.g. the vector processor Cray C-90) could rapidly swap jobs in and out of core. Excess loading slowed the machine, but large jobs could be interactively run/monitored and started with no queue wait. Batch jobs ran at night. Interactive use (usually with wait) is possible only for very small <128ps -30minute test jobs on Seaborg.

Note the ORNL CCS IBM Power4 (cheetah) computer was similarly overloaded this past Summer. Cheetah has not to this point had a formal allocation process. The overload was mainly due to temporary half-machine dedicated use by a special climate project. CCS has been very responsive to Fusion by giving a 3 day age boost to some Fusion users and special help with access to the new Cray X1 (phoenix). The overloading has currently abated somewhat.

Near term cost free recommendations for OFES

While we have found NERSC management generally responsive to our needs, NERSC appears to be highly constrained by MICS policies. OFES needs to obtain more direct control of its share of computing, if MICS cannot be persuaded to change its policies. Here are some suggested changes which should improve the loading efficiency:

(1) Over allocation of Seaborg should stop. In fact yearly total allocations should be less than full capacity (maximum cycles/year less estimated down-time) with some fraction held in reserve. Yearly allocations should be given out quarterly with some small penalty for nonuse in following quarters. This will add to the reserve “bank” of allocation to be given out to active users in succeeding quarters. This will smooth out fluctuations in loading efficiency (E) and will definitely allow it to rise to better than the 50% level. If the wait times decrease significantly and E values across all node count stays above 50%, more of the reserve can be let out to those who have run out of allocation. NERSC aims at 95% utility by full allocation. There should be no doubt that under allocation (with a possibly lower utility) will increase E allowing more productive research.

(2) Priority boosting should be greatly curtailed. If $E > 50\%$ there is little need for it. Users should pay for priority in the premium queue. Temporary and adjusted “age boosts” could be given to priority INCITE projects only to keep their specific E value just above 50%. In no case should “front of the line” boosts be given.

(3) The queue structures should be simplified. There may be a streamlining problem with “square pegs” waiting with only “round holes” available: allowing only full node multiples of 2 (2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048) might help. The endless tinkering with priorities and discount charges to encourage large node count maybe in the end be counter-productive. It may be just forcing users to go where they can’t or don’t want to go. Many runs must be continued from restarts. Previously, Seaborg queues were extended from 24hr to 48hr. It will likely improve E values to allow longer 72hr runs.

(4) The top level Seaborg website should have a daily update of E vs Nodes and W vs Nodes tables. Users could then shift their jobs up and down in number of nodes to get the fastest turn-around time (Nodes x E). This would work especially well with the multiple of 2 node queue structure.

(5) We also believe that the annual NERSC allocation process has become increasingly burdensome (the forms change every year) and irrelevant (with little relation between requests and actual allocation, beyond the rich get richer). Huge amounts of irrelevant technical information about code performance are requested when scientific merit and Fusion program priority should mainly determine allocation. Scientific merit is determined at OFES along with the three year base funding reviews. The allocation process for Fusion’s share should be handled directly by OFES and communicated back to NERSC with minimal consideration of performance (fraction of peak speed, etc.).

Longer term recommendations on new resources.

The US is by far the world leader in Fusion simulations of all types, although it has only half the Fusion funding of either Europe or Japan. New US funding in the Fusion simulation area would have the largest and most cost effective impact on the world Fusion program. The dominance of US Fusion simulation codes could allow the US Fusion scientists to lead and direct the scientific research path for ITER. Inadequate computer capacity (cycles/year) is the key limitation on simulation research productivity. Many computational scientists run through their allocation earlier in each allocation year. (Cheetah coming on line during the past year gave some temporary relief this year.) Only a mix of local clusters or cluster farms to go along with the supercomputers will provide the most cost effective cycles/year. Local clusters managed co-operatively in a small group (instead of open time sharing) can provide cycles on demand with no wait time. Any small computation and theory group who want and can manage a local cluster at no extra cost should have one: *OFES needs to fund these out of the base program.* Local clusters cannot provide community wide time shared computing. Just as with supercomputers, clusters become quickly useless when overloaded. In providing the needed computing tools, DoE needs to redirect its thinking from cycles per second to processors per user. Because of NERSC's origin in MFE computing, and Fusion being the largest single user at NERSC, Fusion is the only major Energy Research division to rely almost totally on MICS supercomputers. Other division (e.g. High Energy Physics, Climate, etc) have much greater access to local clusters funded within the divisions.

In the past MICS has not supported clusters. MICS sees their mission as high performance computing (cycles/sec), and heretofore has been slow to provide the vitally needed computing services. MICS should not be funding local clusters, but MICS should be funding and managing cluster centers with cluster farms, and a mix of all sizes. We know that NERSC does in fact have funding in hand to buy a large cluster to off load the smaller jobs from Seaborg. This is certainly a step in the right direction. We understand the cluster will have 600ps with 3x faster processors than Seaborg, i.e. a 30% increase in cycles per year. Good start, but we need a lot more.

We would go a step further. Since ORNL has apparently won the future 20-100 teraflop mix of high performance (cycles/sec) machines, we think NERSC should take this as an opportunity to lead in computer capacity (cycles/year) and the computing service area. NERSC has the experienced staff in place to do this very well. (The NSF SDSC took the "computing service" rather than the "computer science" strategy from its inception and is arguably the most successful center in the NSF system. It started as a copy of MFE.) At the end of Seaborg life, NERSC could replace it with a large cluster farm, and mix of sizes, all with the same software, uniform management, and maintenance. Various ER sectors and subgroups could have allocated priority use on designated machines. Overhead and staffing economies of scale would be preserved. Phase out and replacement with faster commodity hardware could be done gradually, cluster by cluster (keeping pace with Moore's Law). One could estimate that for the same initial and annual center cost, a cluster center might have 3 times the capacity of a supercomputer center.

Appendix: Statistics from the NERSC database of completed job information.

FY04 = Oct. 1 2003 to Sep. 30 2004

REGULAR CHARGE CLASS (EXCLUDING INCITE JOBS)

SEABORG E VALUE (%) FOR FY2004

ALL

Job Size (No of Nodes)	Q1	Q2	Q3	Q4
1-7	29.6	22.9	19.3	13.6
8-15	34.6	23.0	27.8	8.2
16-31	23.4	13.4	18.5	3.6
32-63	38.9	32.0	30.3	8.9
64-127	29.8	27.8	27.1	3.9
128+	18.9	24.0	25.8	6.1

RUN WALLCLOCK > 8 hours

Job Size (No of Nodes)	Q1	Q2	Q3	Q4
1-7	44.4	36.4	25.9	19.4
8-15	42.2	31.1	38.5	18.1
16-31	34.6	24.9	27.4	5.1
32-63	65.0	61.7	54.0	13.7
64-127	36.8	52.6	66.0	10.9
128+	32.2	76.9	73.9	44.7

For all FY04 Jobs in Regular Charge Classes, excluding INCITE: E = 20.1%
Same as above, jobs using 8 hours or more wallclock: E = 30.2%

SEABORG AVG. WAIT TIMES (HH:MM) FOR FY2004,

ALL

Job Size (No of Nodes)	Q1	Q2	Q3	Q4
1-7	4:46	11:04	9:23	14:31
8-15	1:44	2:21	4:33	23:58
16-31	9:11	12:08	10:22	47:39
32-63	5:38	4:50	7:40	48:35
64-127	7:32	7:32	10:16	71:14
128+	6:01	4:02	5:53	18:46

RUN WALLCLOCK TIME > 8 hours

Job Size (No of Nodes)	Q1	Q2	Q3	Q4
1-7	16:39	22:19	36:28	53:00
8-15	18:21	31:17	22:27	63:14
16-31	24:12	48:28	42:06	208:33
32-63	11:33	15:20	15:21	128:14
64-127	12:32	14:01	11:01	222:32
128+	26:54	6:25	6:08	14:08

E vs WALLCLOCK RUN TIME

48 HOURS (2880m) IS LONGEST ALLOWED RUN TIME

(s=seconds, m=minutes)

WALLCLOCK	N	RAW HOURS (WALL*16*NODES)	AVG. WAIT (HRS)	AVG. WALL (HRS)	E
<15s	31,179 (15.0%)	11,043 (0.03%)	2.70	0.003	0.1%
15-29s	29,569 (14.2%)	13,509 (0.03%)	2.20	0.006	0.3%
30-59s	8,649 (4.1%)	7,961 (0.02%)	3.63	0.014	0.4%
1-2m	6,026 (2.9%)	17,481 (0.04%)	3.53	0.023	0.6%
2-4m	5,897 (2.8%)	40,422 (0.10%)	3.03	0.048	1.6%
4-8m	9,759 (4.7%)	165,614 (0.40%)	5.80	0.092	1.6%
8-16m	14,726 (7.1%)	227,698 (0.55%)	5.85	0.206	3.4%
16-32m	17,570 (8.4%)	538,617 (1.29%)	6.12	0.390	6.0%
32m-64m	18,226 (8.7%)	885,259 (2.12%)	7.10	0.465	6.2%
64-128m	17,197 (8.2%)	1,793,612 (4.30%)	10.22	1.567	13.3%
128-256m	15,582 (7.5%)	3,730,169 (8.94%)	10.21	3.117	23.4%
256-512m	17,171 (8.2%)	9,567,306 (22.93%)	15.55	6.25	28.7%
512-1024m	12,827 (6.2%)	10,107,857 (24.22%)	23.28	11.53	33.0%
1024-2048m	3,974 (1.9%)	10,230,565 (24.52%)	60.00	22.43	27.2%
>2048m	148 (0.1%)	4,392,209 (10.53%)	66.27	45.70	40.8%