SAND2002-1909P

A Recipe Book for Easy Use of the ASCI White Platform

For:

SNL users who need to jumpstart their use of White

v2.1

May 2, 2002

By:

Jim Ang (ed.), Martha Ernest, Kathie Hiebert-Dodd, Curtis Janssen, Jeff Jortner, and Judy Sturtevant

With Additional Input From:

Marty Barnaby, Sue Goudy, Tom Pratt, and Bill Rahe

Change Table

Version	Date	Comments
0.XX	5/25 – 6/20	Early working drafts distributed among the Recipe Book team
1.0	6/20/01	First draft posted on the SNL Apps Top35 Web Site
1.1	7/11/01	Added SEACAS Recommendation to set MP_THREAD_STACKSIZE 10M
1.2	7/23/01	Updates on Compiling 64 bit programs, Processor Performance Measurement, and EnSight SOS Usage on White
1.3	8/03/01	Updates for Nemesis/SEACAS description, default setting for MP_SHARED_MEMORY environment variable
1.4	8/20/01	Updates to EnSight configuration and Pre-Processing for EnSight SOS usage on White, upgrade to 16 GB nodes, reduction of VIEWS partition to 28 nodes, addition of this *Change Table*
1.41	8/22/01	Corrected typo for asciwhite.scf.cln & ice.scf.cln
1.5	11/20/01	Replaced 'switch' with 'switch2' in all the 64-bit mpi references. Added the MP_EUIDEVICE definition, which is also a work-around for the current version of 64-bit mpi and its incompatibility with the double/single hardware.
1.6	3/4/02	Additional information on performance tools
2.0	3/28/02	Added a new section on Grid Services
2.01	4/22/02	Changed page numbers
2.1	5/2/02	Changes to reflect White upgrade from Limited to General Availability. VIEWS partition is back to 32 nodes. Loss of web page integration between the SRN and the other Labs' "Yellow" environments.

Scope

The primary audience for this document are Sandia's code team members who need to get up to speed on how to use the ASCI White platform for the STS Application Milestones. The authors made a conscious effort to limit information to only those essential facts and recommendations that help new users get started.

The White Platform is actually a collection of IBM machines. As shown in the table below the largest partition on the secure side is referred to as ASCI White. The VIEWS partition of White was established to support interactive visualization and data services functions. The Snow machine in the unclassified-restricted environment is a much smaller machine that is often used for initial code development runs and porting. The Ice machine provides priority for interactive over batch jobs. As of April 15, 2002, the entire collection of IBM machines is in General Availability.

Machine Name	Classification	Primary Usage	Nodes-memory
asciwhite.scf.cln	Secure	Large Batch Jobs	461-16GB
VIEWS partition	Secure	Interactive Visualization	32-16GB
snow.llnl.gov	Unclassified	Code Development	16-4GB (8 proc/node)
ice.scf.cln	Secure	Debug & Small Jobs	26-16GB
frost.llnl.gov	Unclassified	Alliance Centers, Science Runs	68-16GB

How to Get Started

This document is based on notes and comments from some of our early users, highlights from LLNL's White documentation and startup/usage advice from some of the technology developers who are providing distance computing capabilities specifically to support distance use of White. Our goal is to provide new SNL users a concise, easy to understand guide. This recipe book is not intended to provide all of the details that users may need or want to learn about. Extensive documentation, tutorials and the latest information are available in the "White Users Manual." This web-based manual has been under development by a team of LLNL, Livermore Computing (LC) user support, technical editors and web page developers for over a year. Links to this documentation are collected in the URLs below.

User Support Points of Contact

There are two established mechanisms for obtaining user support and asking questions about SNL distance computing on ASCI White. If you have a specific question about White you can call the Livermore Computing Hotline (925)422-4531 or send e-mail to *lc-hotline@llnl.gov*. If you have a distance computing question or are not sure if your question is about a SNL or LLNL issue you can send e-mail to *white-help@sandia.gov*. E-mail sent to this reflector is automatically sent to SNL Scientific Computing Department as well as the Livermore Computing Hotline.

Access to more information

We have gathered links to the White Users Manual (Using ASCI White) along with a collection of Distance Computing ASCI User Services and Documentation links on the SRN at:

https://secureweb.sandia.gov/discom/services

and on the SCN at:

https://www-isn.sandia.gov/discom/services

Both of these links will query for your SNL userID and Kerberos password.

Establishing cross-cell accounts: White and SecureNet Classified Guest Accounts

SNL users need two accounts to login to White. First you need to have an account on ASCI White. As of 4/15/02, there are no longer any limitations on the number of White accounts per laboratory. Sandians who request an account for White will automatically also be requesting an Ice account. The second account you need is a SecureNet Classified Guest account. If you have been doing ASCI work on SK&Y or Blue Mountain, it is likely that you already have one of these accounts established.

The SRN and SCN URLs above provide links to request these accounts.

Establishing cross-cell accounts: Snow and Intersite Unclassified Guest Accounts

The ASCI Snow platform is a small, unclassified version of White at LLNL. While this system is much smaller and tends to be less stable than White, you may find it worthwhile to establish an account on Snow as well. There is no limitation on the number of Snow accounts per lab. A useful capability for users with accounts on both White and Snow is the File Interchange Service (FIS). This is a service for copying files from LLNL's open (Snow) to closed (White) environment. Note: FIS requires a local LLNL account, i.e., your cross-cell credentials are not accepted by this service.

Resource Status URLs & monitoring tools, etc.

The URLs given above also provide links to various resource status and monitoring pages. You will find links to status and monitoring of Tri-Lab resources from the LLNL perspective and from the SNL perspective. The SNL perspective is found in the Sandia's What's Up in the TriLab link and this page will query for a special userID/Password of "guest"/ "guest".

New firewalls between the Tri-Labs have broken the integration between the SRN and the other labs' "Yellow" environments. Therefore, ***you can no longer*** access LLNL status and monitoring page by entering your fully qualified DCE username, i.e., /.../dce.sandia.gov/<userID> and entering your SNL Kerberos password in the password field. You should still be able to access this information from the links in the SCN URL given above.

Allocation and Access to Resources

The ASCI White platform consists of 512 – 16 processor nodes divided into 4 interactive and I/O, 16 GPFS server, 32 VIEWS and 460 batch nodes. As of 7/25/01 all nodes have 16 GB of memory. The Snow machine consists of 1 interactive and I/O node and 15 batch nodes. On the Snow machine each node has 8 processors with a total of 4 GB of memory.

Batch nodes

The batch nodes of ASCI White are allocated among the three labs through the automated scheduling and bank accounting mechanisms provided by LLNL's Distributed Production Control System (DPCS). Sandia's allocation is set at 128 nodes, however there are a number of constraints that place additional limitations on the number, size, and duration of jobs that are submitted to the batch queue. The allocation of bank points within Sandia is done under the oversight of Sandia's ASCI Applications Program Lead, Mike McGlaun. Users who wish, may submit jobs that exceed Sandia's 128 nodes limit by using a standby queue. However, because these jobs encroach on another lab's allocation of nodes, they are killable. Because DPCS does not have a provision for coordinating the submission of high priority, large, long running jobs

among the three labs, a regular VTC meeting and white-epr@llnl.gov e-mail reflector is used to facilitate communication/coordination among representatives from the three labs.[1]

Interactive VIEWS partition

The VIEWS interactive partition consists of 32 nodes for visualization, post-processing and other data services tools that are used in an interactive manner. To support interactive usage, this partition is not subject to the DPCS "fair share" access/allocation mechanism that is used for the batch partition of White. The use of the VIEWS partition is governed by "good-citizen" behavior among the Tri-Lab user community. As such this means that prior to submitting a job that uses > 8 of the available 32 nodes, it is common courtesy to send an e-mail message to the white-views@llnl.gov e-mail reflector to give other VIEWS partition users a heads up on how many nodes you are interested in using, for how long and how you can be reached in an emergency. Obviously the greater the lead time you can provide the better.

Storage

The White platform has ~60 TB of general parallel file system storage (GPFS) in /p/gw1. This is the recommended location for all parallel I/O. The specific user's file space is /p/gw1/<userID>/. GPFS is not backed up and is subject to purging. Long term archiving of data may be performed on LLNL's HPSS or by moving files back to SNL. To support the movement of files back to SNL a number of tools are available and described in the parallel file transfer section below.

Logging into LLNL platforms

How to Access White and Snow

The cross-cell account allows you to perform your DCE/Kerberos authentication here at SNL, obtain a forward-able DCE credential to access and use resources at LLNL and some of our Tri-Lab integrated network applications.

To login to ASCI White, start with the following commands on your local machine:

```
% /usr/local/bin/kinit -f
% ssh asciwhite.scf.cln
```

If you have an account on snow you would login to snow in an analogous manner.

```
% /usr/local/bin/kinit -f
% ssh snow.llnl.gov
```

White Simulation Development Environment

Many SDE Tools are listed in the White User Manual. This document lists only a small subset to get you started. If one of these tools does not work for you there may be another that will.[2]

---

[1] If you have issues or concerns SNL needs to raise at the White EPR meeting call Jim Ang at 844-0068 or Paul Nielan at 294-2510.

[2] If you have questions about the current Simulation Development Environment and would like to discuss other options call Curtis Janssen at 294-1509.

Paths and Environment

The paths and environment are set up automatically when you login. For more information, C-shell users should look in /etc/csh.login and Bourne-shell users can consult /etc/profile.

Shared Areas for Projects/Users

In the LLNL environment there is a file structure to accommodate shared user/project files. Sandia application code executables can be found in subdirectories under /usr/gapps/. This path is the ASCI White equivalent of /Net/projects/ on janus-s. To request the creation of a shared area or to look at the policy for this area click on the following URL: https://www.llnl.gov/lcforms/usr_gapps.html . At SNL our computer account coordinators are Walt Vandevender, Paul Nielan, Matt Leininger, and Dave Davis. After completing, printing, and signing (you and your local site coordinator) the form, submit it via fax or mail to the LC Customer Services and Support Group.

Group Accounts

To request creation of a new group, or to add someone to an existing group, fill out the Create/Update Group form. This form can be obtained at URL: https://www.llnl.gov/lcforms/group.html. Note: when filling out the request forms try leaving the LLNL Employee Number and L-Code blank, since they are not relevant for SNL users. If these are required fields (and they originally were) put in numbers and cross them out later before submitting the form. After completing the form, print it, obtain the required signatures, then send it to the LC Hotline via fax or mail. After it is received, the requested group will be created and the group owner will be notified when it is ready to be used.

Compilers

Two front-ends for each compiler are available. The serial front-end is used to compile code for single processor, interactive runs. The MPI front-end makes sure that the MPI libraries are linked in. Such executables must be run with poe. Furthermore, several versions of each compiler are available. Older versions of each compiler can be accessed by prefixing the compile command with 'old'. Newer versions have the 'new' prefix. On a regular basis the compilers are rotated so that the default compiler becomes 'old' and the 'new' compiler becomes the default.

The IBM C++ compilers should provide full support for ISO standard C++ except for the "export" statement. The IBM C++ compiler is recommended, but if you have problems with the IBM compilers, consider using the KAI compiler. The KAI compiler is required if you wish to use OpenMP with C++. More information on the KAI compilers is available at: http://www.kai.com/C_plus_plus/Current/doc.

There are other compilers and other versions which are listed in the 'Compilers' section on the ASCI White WWW pages: http://www.llnl.gov/asci/platforms/white/software

The primary compilers are listed in the table below.

Language	Serial Compiler Name	Parallel Compiler Name
C (IBM)	xlc_r	mpcc
C++ (IBM)	xlC_r	mpCC
Fortran 77 (IBM)	xlf_r	mpxlf
Fortran 90 (IBM)	xlf90_r	mpxlf90
C++ (KAI)	KCC	mpKCC

Common Compiler Options

As usual, the '-O' and '-g' options can be respectively given for optimization and generating debugging information. Both can be given at the same time, but this can lead to misleading information from debugging tools.

Common Linker Options

To allow a heap size > 256MB, use '-bmaxdata:0x70000000'. To use the maximum stack size of 256MB, use '-bmaxstack:0x10000000'.

Large File Support

Programs compiled using the IBM C and C++ compilers can manipulate files > 4GB by specifying '-D_LARGE_FILES' during the compilation. This will work for 32 bit programs. For 64 bit programs this flag is unnecessary as large file support is automatically enabled.

Compiling 64-bit Programs

To generate and link 64-bit code with the IBM compilers, give the compilers the '-q64' option. When archives containing 64-bit objects are manipulated with commands such as 'ar' or 'nm', the '-X 64' option is required. To link 64 bit executables with the 'ld' command, the '-b64' option must be given. In place of the '-q64' and '-b64' options, the OBJECT_MODE environmental variable can be set to '64'.

The 64-bit version of MPI is still experimental—it is not intended for production use. Read '/usr/local/lpp/64bitmpi/switch2/ppe.poe/README.64bit.prpq' for more information. To use 64-bit MPI, set the environmental variable 'MP_PREFIX' to '/usr/local/lpp/64bitmpi/switch2' to allow the mp compiler scripts to locate the current include files and libraries. 'MP_PREFIX' only works with mpcc and mpCC, to use Fortran you'll need to use the serial compiler and provide the necessary include paths, library paths, library names, and other options as follows:

```
% xlf -q64 -O \
  -I/usr/local/lpp/64bitmpi/switch2/ppe.poe/include/thread64 \
  -c program.f
% xlf -q64 -o program program.o -binitfini:poe_remote_main \
  -L/usr/local/lpp/64bitmpi/switch2/ppe.poe/lib -lmpi_r
```

When running a 64 bit MPI application, it is essential that the environmental variable MP_EUILIBPATH is set to '/usr/local/lpp/64bitmpi/switch2/ppe.poe/lib'.

Libraries

BLAS: Use the optimized routines from the ESSL library by linking with '-lessl'. Do not use '/usr/lib/libblas.a'.

LAPACK: The ESSL library provides optimized LAPACK routines; however, a few of the LAPACK routines are missing. Link with ESSL by using '-lessl'.

Finding General Bugs

TotalView is a full featured parallel debugger. The White User Manual should be consulted for complete information and tutorials. To quickly get up and running with TotalView, type 'totalview poe –a *program_name program_arguments*'. This will bring up two windows. In the 'poe' window

type 'g' (for go). A dialog will popup asking if you want to stop parallel tasks before they enter MAIN. Click on 'yes'. A window with the main routine will pop up. You may now set breakpoints by left clicking on line numbers. Type 'g' again, and examine data when the breakpoints are reached.

If your program is producing invalid floating point numbers (NaN's), then you may want to recompile, enabling trapping of bad results near where they first occur. This is done with the compiler option '-qflttrap=overflow:zerodivide:invalid:enable'. Then use TotalView to find where the trap occurs.

Finding Memory Bugs

Zerofault can find illegal memory references and leaks. In parallel runs, 'zf_id' is used to collect zerofault information from one node in a file. 'zf_ui' will bring up a window displaying that information. If the executable is named 'exec', zerofault can be used to analyze task 0 as follows:

```
% poe zf_id 0 exec
% zf_ui exec.zfo
```

MPI Performance Measurement

The mpiP package performs lightweight MPI performance measurement. The user guide is in /usr/local/mpiP. It is used by linking with the following flags: '-L/usr/local/mpiP/lib –lmpiP'. MPI_Finalize will cause an output file to be automatically written to the current directory.

Vampir is a tool that can provide detailed graphical information about MPI communication. A beta version of Vampir, provided by ASCI PathForward, is available on White and Snow. This version provides scalability enhancements and is thus the recommended version. To use Vampir, the program must be relinked with the Vampirtrace library which will write a trace file at the end of a run. Vampirtrace files can be analyzed with the Vampir GUI. Vampir is installed in /usr/local/kppp and the documentation can be found in the doc subdirectory. The version on Snow is typically more up to date than the version on White, so it is likely to work better.

Here is a quick recipe for using Vampir:

```
setenv PAL_LICENSEFILE
/usr/local/kppp/etc/license.dat
mpCC -o prog prog.o -L/usr/local/kppp/lib -lVT -lld
poe prog
/usr/local/kppp/bin/vampir prog.stf
```

Processor Performance Measurement—Hardware Performance Counters

The HPM package can be used to collect hardware performance data from program runs. It is not necessary to modify or even recompile the program to collect HPM information. The HPM software is installed in /usr/local/src/HPM/HPM_V2.3 where, in the doc subdirectory, more information can be found. To simplify HPM data collection and processing for parallel runs, the hpmjoin and hpmwrap scripts in /usr/gapps/vprof/hpm can be used as follows:

```
poe /usr/gapps/vprof/hpm/hpmwrap executable arguments
/usr/gapps/vprof/hpm/hpmjoin hpm.*
```

The hpmwrap script ensures that the results from each MPI task is written to a separate file and hpmjoin will sum the results from all these files.

The hpmwrap script ensures that the results from each MPI task is written to a separate file and hpmjoin will sum the results from all these files.

```
#include <libhpm.H>
void solve() {
  hpmStart(1,"solve");
  ...
  hpmStop(1);
}
main() {
  hpmInit(mpi_rank,"codename");
  ...
  solve();
  hpmTerminate(mpi_rank);
}
```

In this case, the code must be compiled with "-I/usr/local/src/HPM/HPM_V2.3/include" and linked with "-L/usr/local/src/HPM/HPM_V2.3 -lhpm -lpmapi". The hpmwrap script should not be used. The hpmjoin script is still useful for summarizing information from all the nodes. To profile more than one code region in a single run, consult the HPM documentation. The hpmjoin script will sum together the results from all profiled code regions.

Processor Performance Measurement—Execution Time Profiling

The gprof utility can be used to report the time spent in each routine along with other information. First, the code must be compiled and linked with the '-pg' option. Then, when the program is run, a gmon.out file will be produced. For parallel runs, a gmon.out file will be produced for each MPI task. Such files can be analyzed by executing 'gprof gmon.out > gprof.out'.

VProf provides line by line execution profiling. It is available on snow and white (for access to the software on snow you must be added to the vprof group[3]). VProf requires only relinking of an application that has been compiled with normal optimization options and the "-g" option. Information on using VProf is available in the directory /usr/gapps/vprof/current. README gives a general description of VProf and README.AIX gives information specific to the IBM SP environment.

Running Jobs on LLNL Platforms

Quick guide to running jobs (interactive or batch)

For all the options see the following URL: http://www.llnl.gov/asci/platforms/white/index.html
Then select the "Running Jobs" button located on the left frame of that web page.

Set some combination of the following POE environment variables
(or use corresponding command line argument).

-procs	MP_PROCS
-nodes	MP_NODES

[3]To access VProf, or, if you have questions, suggestions, or bug reports for VProf, contact Curtis Janssen at 925-294-1509.

-tasks_per_node	MP_TASKS_PER_NODE
-shared_memory	MP_SHARED_MEMORY
-rmpool	MP_RMPOOL

MP_PROCS Determines the number of program tasks. Valid values are any number from 1 to 4096. If not set, the default is 1. The value of this environment variable can be overridden using the -procs flag.

MP_NODES Specifies the number of physical nodes on which to run the parallel tasks. It may be used alone or in conjunction with MP_TASKS_PER_NODE and/or MP_PROCS. The value of this environment variable can be overridden using the -nodes flag.

MP_TASKS_PER_NODE Specifies the number of tasks to be run on each of the physical nodes. It may be used in conjunction with MP_NODES and/or MP_PROCS, but may not be used alone. The value of this environment variable can be overridden using the -tasks_per_node flag.

MP_SHARED_MEMORY Determines whether or not tasks running on the same node should use shared memory (instead of the SP switch) for message passing. Valid values are yes and no. If not set, the default is yes (changed on 7/25/01). The value of this environment variable can be overridden using the -shared_memory flag.

MP_RMPOOL Specifies which pool of nodes should be used to run the job. To run in the DEBUG pool, specify 0. Generally, other pool numbers are reserved for batch use only.

Other useful variables:

MP_NEWJOB Determines whether or not the Partition Manager maintains your partition for multiple job steps. Valid values are yes or no. If not set, the default is no. The value of this environment variable can be overridden using the -newjob flag.

MP_IONODEFILE The name of a parallel I/O node file -- a text file that lists the nodes that should be handling parallel I/O. This enables you to limit the number of nodes that participate in parallel I/O, guarantee that all I/O operations are performed on the same node, and so on. Valid values are any relative or full path name. If not specified, all nodes will participate in parallel I/O operations. The value of this environment variable can be overridden using the -ionodefile command-line flag.

Interactive job examples using POE

Example 1: The following executed and wrote stdout to the screen.

```
snow01{jesturt}86: poe -rmpool 0 -nodes 1 -tasks_per_node 1
0031-503  Enter program name and flags for each node
../../nem_join/nem_join
```

Example 2: The following is a script used to set environmental variables and then use POE commands to run CTH interactively.

```
#!/bin/csh
```

```
#
# sEFP3d : run cth interactively
#
set ID = a
#
set POE_FLAGS = "-rmpool 0 -nodes 1 -tasks_per_node 8"
#
cd $HOME/Work/Test/EFP3d
setenv CTHPATH $HOME/Work/cth
setenv NUMRAIDS 1
setenv RAID1 /p/gw1/spgoudy/Output
poe $CTHPATH/bin/aix/mpicthgen i=efp3d.in id=$ID $POE_FLAGS
poe $CTHPATH/bin/aix/mpicth    i=efp3d.in id=$ID $POE_FLAGS
```

Batch Job Examples - DPCS

Example 1: Script to submit CTH job to snow via the batch system DPCS. Note how the output is directed to the GPFS area on gw1.

```
#!/bin/csh
#PSUB -c snow
#PSUB -ln 1
#PSUB -g 8
#
# pEFP3d : run cth in batch
#
set ID = A
#
cd $HOME/Work/Test/EFP3d
setenv CTHPATH $HOME/Work/cth
setenv NUMRAIDS 1
setenv RAID1 /p/gw1/spgoudy/Output
poe $CTHPATH/bin/aix/mpicthgen i=efp3d.in id=$ID
poe $CTHPATH/bin/aix/mpicth    i=efp3d.in id=$ID
```

Example 2: Generic script using POE directives

```
#!/bin/csh
#
# Batch job shell script example:
#
set POE_FLAGS = "-nodes 1 -tasks_per_node 8 -shared_memory"
#
# To run in the debug pool
#set POE_FLAGS = "-nodes 1 -tasks_per_node 8 -shared_memory -rmpool 0"
#
poe ./<my_executable> $POE_FLAGS
```

Example 3: Generic script using PSUB directives

```
#!/bin/csh
#
# Batch job shell script example:
#
# PSUB -c white
```

```
# PSUB -r myjob      # Set job-name to myjob
#
# PSUB -ln 1         # 1 nodes
# PSUB -g 8          # 8 tasks
# PSUB -noDFS        # Not accessing DFS; no need to manage DCE
credentials.
# PSUB               # No more embedded options.
#
poe ./<my executable> $POE_FLAGS
```

psub parameters

-g [tasks][switch][@layout]

Geometry is used in conjunction with the -ln option to determine how tasks will be assigned to nodes. The tasks value is the total number of tasks that will be assigned to the job. The switch value is the communications switch type and may be either "ip" or "us". Layout may be either "tpn<number>" where <number> is the number of tasks per node or "dist" which specifies that tasks should be spread as evenly as possible among nodes. The default switch is "us" and the default layout is "dist". If tasks is not specified, then if the layout is "dist", the number of tasks is equal to the number of nodes; but if the layout is "tpn<number>", then the number of tasks is the multiple of the number of nodes requested and the <number> of tasks per node specified. If both tasks are specified and the layout is specified as "tpn<number>", then the number of tasks per node multiplied by the number of nodes requested must equal the number of tasks specified.

-ln nodecount

Sets the number of nodes required for a parallel job. If this option is not specified, the job will be executed as a single node serial job. If "-ln 1" is specified, the job will be executed as a single node parallel job. The value given for the -ln option must be greater than 0. The -ln option is appropriate either when the job is to be run on a MPP (Massively Parallel Processor) machine, such as an IBM SP or for MPI jobs that are to be run on an SMP (Shared Memory Processor) architecture such as a COMPAQ cluster that supports gang scheduling. (see section on MPI runs.)

-noDFS

Specifies that the users job will not access DFS and that there is no need to manage DCE credentials for this job.

Grid Services

The DisCom^2 Grid Services project provides a software infrastructure that enables the use of ASCI's High Performance Computing resources, regardless of location, operating system, or job control system. For this version of the Recipe Book, one service available from a user's workstation will be described. This service is the web-based Resource Monitoring Services, which provides information about all the ASCI platforms.

The Grid Services project also provides services to help the user create, modify, and submit a workflow (a sequence of file transfer and computational steps) to White or any of the ASCI platforms. The Production Wizard (PW) application is a general-purpose method to submit workflows. Originally, how to use PW was to be included in this document. However, due to the complexities of the ASCI Tri-lab classified computing environment, we ask the user to contact the

Grid Services team for help in getting PW installed, their environment properly configured and initial workflows created. These complexities are being understood and reduced by current efforts and a future version of this cookbook will provide a more complete set of instructions. A short description of PW is included. PW is only one way for a user to submit jobs/workflows. Applications can be "grid enabled" directly, but describing that capability is beyond the scope of this document.

More information about Grid Services can be found in the ASCI Grid Services User's Guide for the November 2001 Capability Release located on the SRN at http://gridservices.sandia.gov/ (click on the "User Documentation" link) and on the SCN at https://www-isn.sandia.gov (click on the "Grid Services" link under Engineering Tools). If you have questions or would like to use the Grid Services to submit workflows/jobs to White or other ASCI resources, please contact Grid Services Support at (505) 844-2569 (Kathie Hiebert-Dodd) or e-mail dm-help@sandia.gov.

Using Resource Monitoring Services

The Web-based Resource Monitoring Services displays grid, resource, queue and job status information from a Web Browser (Netscape 4.6 or later is preferred). In this section, the following specific Web pages will be described: Grid Utilization, User Job Status, Queue Status, and Node Counts by Bank/Queue and Class. The information for these pages is collected from all the tri-lab sites; therefore it might take as much as 20 seconds to load.

At SNL, from your classified desktop, first go to the Sandia Classified Home Page: https://www-isn.sandia.gov, then click on the Grid Services link under "Engineering Tools." This will position you at the following links on the Engineering Tools page:

- Grid Services – takes you to the **Grid Services** - Entry Page
- Grid Monitoring Information – takes you directly to the **Grid Utilization** page

When you click either link you will again be prompted to enter your Kerberos password for authentication. Anyone with an account on White (or any ASCI classified resources) will be able to view this information. If this link doesn't work, please contact Grid Services Support at (505) 844-2569 (Kathie Hiebert-Dodd) or e-mail dm-help@sandia.gov. There have been situations where internal firewalls block access to the Resource Monitoring Services page.

Grid Services – Entry Page

The Grid Services entry page provides users with a link to the "Grid Utilization" monitoring page. This page shows all ASCI Resources and their current levels of utilization along with the users running jobs on each resource. (See the next sections for details.)

The "Entry" page also contains:

1. General documentation with some background on the computational grid;
2. Link to the Tri-lab Resources and Services, an ASCI Machines Status page that provides information about what resources are currently available.
3. (This will require another authentication, be sure to enter your fully qualified DCE userID. For example, from SNL use /.../dce.sandia.gov/<userID>);
4. The complete User Manual for Grid Services;
5. Information for Grid Administrators, such as links to State-of-Health, Resource Information, and the DRM Administrators Guide.

ASCI Grid Utilization Page

Use the Grid Utilization page to monitor the status of your computing jobs, whether or not they were submitted through the Grid Services. Grid Utilization provides a summary view of the grid with current status of compute resources at SNL, LANL, and LLNL.

Tabular View:

- Lists each compute server with a batch queuing service or active jobs.
- Lists the total and active CPU counts and number of jobs pending and active.

User Selection

It is possible to filter the user jobs that are displayed to reduce the volume of data. Click on the desired user ids in the **Select Users** box. (Use the Control Key on a PC, or the ALT Key on a Unix Workstation, while clicking to select multiple user ids.)

Two options are available for new displays:

- Click the **Refresh** button to filter the selected users and refresh Grid Utilization; or
- Click the **Job Detail** button to display pending and active jobs for the selected users.

Auto Reload

The Grid Utilization information is dynamically updated approximately every 30 seconds, so there is an option to refresh the information on this web page.

- Click the **Auto Reload** link to see a "Help" window.
- Click the **Checkbox** and then the **Refresh** button to turn on automatic reloading.

Graphical View

- Graphs the number of Busy & Available CPU Nodes, categorized by Site & Machine.
- Adds detail about the utilization of CPUs within the queue allocation with percentages.
- Graphs active and pending jobs for all users with the option to show only selected user jobs.

The Graphical View also provides three links which will pop up a separate browser window with selected details:

- Click on any user id link to view the **Job Status** detail for all jobs submitted by the selected user.
- Click on any queue name to view the **Queue Status** page with job details for all selected queues.
- Click on any machine name to view the **Node Counts by Bank/Queue and Class** for the selected resource.

Click the **Multiple Popups** checkbox and each hyperlink will open a new browser window to facilitate comparison. This information is being collected from the three labs; therefore, it might take as long as 20 seconds for any one of these pages to load.

User Job Status

The User Job Status page appears after clicking any userID listed in the graphical portion of **Grid Utilization** page or by selecting multiple userIDs in the selection box on the left and clicking the **Job Detail** button. The User Job Status page reports all jobs across the grid, including multiple sites, machines and queues.

This web page can answer the following questions:

- On which resources do I have pending or active jobs?
- How long has my job been waiting in the queue?
- Which jobs have the highest priority?
- What is the more detailed status of my pending and active jobs?

Click the **Auto Reload** checkbox to have the page periodically load new statistics, or click the **Refresh** button to load new data on demand.

Queue Status

The Queue Status page appears after clicking on any queue name listed in the graphical portion of **Grid Utilization** page. Select different queues and click the **Refresh** button to change the contents of the Queue Status page. The Queue Status web page can answer the following questions:

- What is my job's position in the queue in relation to other pending jobs?
- How long has my job been waiting in the queue?
- Which jobs have the highest priority?
- What are the attributes and status of each queue?

Click the **Auto Reload** checkbox to have the page periodically load new statistics, or click the **Refresh** button to load new data on demand.

Node Count by Bank/Queue and Class

The Node Count by Bank/Queue and Class page appears after clicking the Resource Machine name listed in the graphical portion of **Grid Utilization** page. You may examine the CPUs and Node Counts for selected jobs with Subtotals by Bank/Queue & Class and the Totals for the current selection.

Bank/Queue – ASCI White will show Bank – other resources will provide Queue name
Class – Lists the jobclass of the submitted job where available
CPU Count – Number of CPUs Requested – that determines the nodes consumed
Node Count – Number of Nodes (May be calculated from CPUs/Node for resource)
Priority – Numbers between 0 and 1, the larger numbers mean higher priority
Completion Estm – Estimated Time of Completion

Select any number of Bank/Queue names in the selection box and click **Refresh** to filter job list.

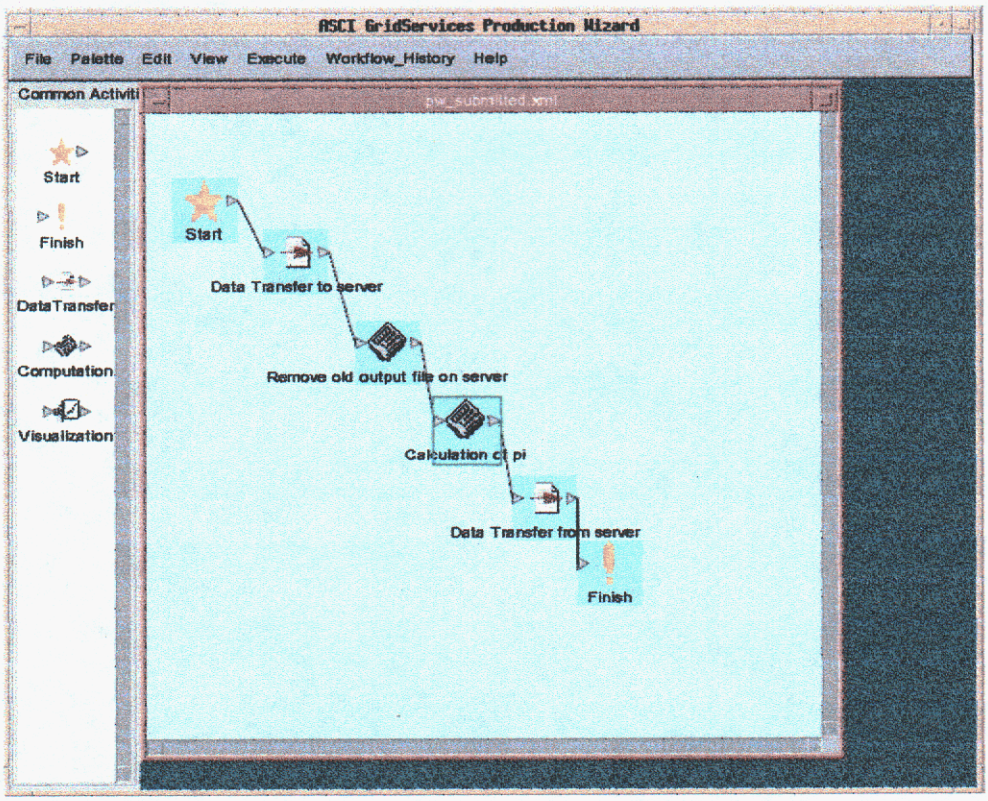
Enter a number in **# Nodes Limit** box and click **Refresh** – the Subtotals line will turn RED when any selected Bank & Class goes over the Node Limit specified.

Click the **Auto Reload** checkbox to have the page periodically load new statistics, or click the **Refresh** button to load new data on demand.

The Production Wizard (PW) Overview

The PW is the generic user interface to the DisCom^2 Grid Services for submitting and monitoring a workflow. PW uses the user's account name and Kerberos authentication credentials to connect securely to the ASCI Tri-Lab Classified Grid. Through PW, the user can open a workflow definition that describes a series of file transfer and computation steps. The user may alter parts of the workflow definition from a graphical display of the workflow-steps, performing such changes as specifying computation resources, locating files to be copied across the Grid, setting the order in which the workflow steps are executed, entering input parameters for computations, and providing names of files which hold input data for or capture output data from computations. From PW, the user can submit the workflow for execution and monitor the execution of batch jobs at each step, examining output from the steps or error messages indicating problems along the way. A completed workflow can be saved easily, thus recording for review any time in the future the input data sources, the produced output data filenames, and the complete record of execution of the steps in the workflow.

The PW workflow main screen is shown in the following figure:



Production Wizard Workflow Main Screen

Getting Help with Grid Services

Please call the Grid Services Support at (505) 844-2569 (Kathie Hiebert-Dodd) or send e-mail to drm-help@sandia.gov.

Pre & Post Processing Data Services

At the present time there are three recommendations for post-processing and visualizing your data.

1. Manipulate the data into manageable (but large) data sets and move these data sets over the parallel WAN to Edison or your local server for visualization.
2. Manipulate the data on White: move data to or access data from the Interactive Visualization partition of White for further manipulation/rendering/visualization/etc. The amount of post processing done could be different for each run regardless of the application. It becomes dependent upon what is most efficient for the current run, the output data characteristics, the speed of the WAN, and the capabilities in place for remote visualization.
3. Transfer all data to Edison or your local viz server before any post processing.

Note: The Wide Area Network is tuned for larger file sizes. Each file transferred to a machine causes a file to be opened/created and there is an overhead to open/create a file. On the HPSS this overhead can be prohibitive when moving thousands of small files. To manage this overhead, we recommend the use of post-processing tools on White such as nem_join.

Nem_join may be used to "join" a large data set with many files into a small number of large files. Subsets may be "joined" based on lists and/or ranges of time indices (time steps), lists of nodal variables, and lists of elemental variables. See Nemesis tools below.

SNL's SEACAS Environment

Users need to be members of the 'seacas' group.[4] Contact Thomas Otahal to get added to the group.

For 32-bit SEACAS, set the ACCESS environment variable to:
/usr/gapps/seacas/aix_4/current

For 64-bit SEACAS, set the ACCESS environment variable to:
/usr/gapps/seacas/aix_4/current64

To execute 64-bit SEACAS tools/apps in parallel, you must set the following environment variables or use corresponding command line arguments:
MP_PREFIX /usr/local/lpp/64bitmpi/switch2
MP_EUILIBPATH /usr/local/lpp/64bitmpi/switch2/ppe.poe/lib
MP_EUIDEVICE css0

Recommended are the following settings for 32 or 64 bit execution:
MP_EUILIB us
MP_SHARED_MEMORY yes
MP_THREAD_STACKSIZE 10M

Nemesis tools currently available as follows:
nem_slice 32-bit/64-bit serial
nem_slice_salinas 32-bit/64-bit serial (contains modifications for Salinas)
nem_spread 32-bit/64-bit serial/parallel
nem_join 32-bit/64-bit serial/parallel

Note: Do not execute the nemesis tools on the service nodes (login nodes). The tools require large amounts of memory for data and MPI buffers. Sharing the memory with other users, by working on the service nodes, is NOT recommended. Our recommendation is to run nemesis tools on the VIEWS partition. All 32 bit and 64 bit SEACAS tools and libraries are working correctly.[5]

For more information on all SEACAS tools and libraries see the SEACAS web page on the SRN:
http://sass2248.jal.sandia.gov/SEACAS/Documentation/SEACAS.html

Remote Visualization

Usage of Interactive Views Nodes

Each of the VIEWS' nodes has 16GB of memory, 16 processors, and a single I/O channel. When using these nodes for data services and visualization, one must balance the competing needs of

---

[4] For general SEACAS support on white, please contact Thomas Otahal at 844-9575 or Judy Sturtevant at 845-9448.

[5] If you have questions about the nemesis tools or have problems with the 64 bit tools and libraries call Judy Sturtevant at 845-9448.

memory usage and I/O. The following are suggested guidelines for the best utilization of resources when running various programs.[6]

EnSight configuration on white

The following settings are needed to run EnSight on the white machine.

```
setenv ENSIGHT7_HOME /usr/local/EnSight/7.3
setenv ENSIGHT7_ARCH ibm_4.3_64
set path = ( $ENSIGHT7_HOME/bin $path)
setenv ENSIGHT7_MAX_THREADS 4
```

Pre-processing for EnSight SOS usage on White

For large datasets you will utilize the EnSight Server of Servers (SOS). The EnSight SOS differs from EnSight Gold in the following:

- The EnSight client communicates with the EnSight SOS which in turn talks to multiple EnSight servers.
- The EnSight SOS has a subset of the features available in EnSight Gold.

Many of the Sandia codes running on white generate 100s to 1000s of files. EnSight SOS does not scale to this number of input data files and it is therefore required to reduce the number of input data files. The sierra-concat program is used for this purpose. You will typically want to reduce the number of input data files to a range of 8-32 files depending on the data file sizes.

Perl scripts have been developed to simplify the conversion process. Before using the scripts it is suggested that you create a new directory to hold the configuration and output files.
You will use one script to generate sierra-concat commands and then another to generate an EnSight SOS case file. For the sierra-concat conversion, you will first need to create a configuration input file for the conversion with the following format (the colons are important):

Total number of files	:1024	
Number of output files	:16	
Input file extension	:e	
Output file extension	:exo	
Input file directory	:/p/gw1/jnjortns	← (by convention, your userID is the directory name)
Output file directory	:/p/gw1/jnjortns/new	
Input file basename	:g1s1	
Script file basename	:sierra-gen	

This file specifies that 1024 parallel output files of the name g1s1.e.1024.** will be reduced to 16 files with the names g1s1.exo.16.**. You can get a sample copy of this file at /usr/gapps/seacas/ensight/sc.inp. Once you have a configuration file (g1s1.inp), you will then issue the following command: sc-gen g1s1.inp

This will generate multiple files. You will get a variable number of files like sierra-gen.00, sierra-gen.01, etc. depending on the number of input and output files. Each of these files will have sierra-concat commands in them. You will also get other files called "sc-cmd.**" and "sc.run". The "sc.cmd.**" files batch together the sierra-gen files and the "sc.run" file contains the names of the sc-cmd.** files and will be used to execute them. To execute the script, do the following:

---

[6] If you have questions about distance visualization from ASCI White to SNL call Jeff Jortner at 294-3846.

```
set POE_FLAGS = "-rmpool 2 –nodes # –pgmmodel mpmd –savehostfile proc.list
```
(number of nodes is set to number of sgen files)
```
poe –cmdfile sc.run  $POE_FLAGS >& sc.log &
```

This will execute the job in the background and you can logoff without killing it. When the job is finished, you will have a reduced number of parallel output files.

EnSight SOS usage on White

Usage of the SOS requires a SOS input data file (typically xxx.sos). This data file specifies where each EnSight server will run and where it's input exodus file resides. As this file needs specific machine names, it must be generated after you have interactively allocated nodes in the Views partition. You will need to create another configuration file for generating the SOS file.

Server Type	:exodusII_gold
Number of servers	:32
Number of servers per node	:4
Host List filename	:/p/gw1/jnjortns/data/sos.host
EnSight Server executable	:usr/local/EnSight/7.3.1/ensight73/bin/ensight7.server
Data file path	:/p/gw1/jnjortns/data
Case file basename	:rect_large.out.e.32
Output SOS filename	:/g/pw1/jnjortn/data/rect_large.sos

You can get a sample copy of this file from /usr/gapps/seacas/ensight/sos.inp. You will then issue the command sos-gen sos.inp. This will automatically allocate an appropriate number of interactive Views nodes, start an xterm in each one of them, and display them on your terminal. Your xxx.sos file has now also been generated.

You can now start up an EnSight client on Edison by issuing the following command: ensight7.client –cm –security xxx, where xxx is any number. If you do not specify a number one will be generated for you. You can now start up the EnSight SOS on a **login** node of White (you get less network bandwidth if you run the SOS on a compute node) with the following command: ensight7.sos –c yyyy –security xxx, where xxx is the number specified or generated previously and yyyy is one of the following:

edison.scn.sandia.gov
172.16.90.4
172.16.91.4
172.16.92.4
172.16.93.4

In almost all cases you should specify one of the IP addresses. These paths will use the high speed connections between LLNL and SNL. If you specify edison , then you will be using the Securenet connection between LLNL and SNL.

You can now use EnSight to load your SOS input file. Make sure to specify exodus_gold as the data format.

When finished running EnSight you can type "q" in each of the xterms to close them and then logoff.

How to perform file movement

Currently, in order to transfer data from LLNL White you must be logged into and issue the pftp commands from the White platform. It is anticipated that in the near future the server software

will be installed on White, then you will be able to issue pftp commands from Edison (and other machines) to transfer data to/from White.

On the white nodes, scripts have been placed in the directory /usr/local/bin. This path should be in every users PATH... Executing these scripts initiates the Parallel FTP Client in interactive mode. Below are the script names currently on White.

pftp2edison	Establishes an FTP session to transfer data to edison at SNL-NM
pftp2janus	Establishes an FTP session to transfer data to janus-s at SNL-NM
pftp2snl	Establishes an FTP session to transfer data to HPSS at SNL-NM

Information on use of the Parallel FTP Client may be obtained by typing man pftp_client.

These scripts set several parameters/flags to specify window sizes, ports, block sizes, etc. These parameters have been determined via testing and tuning of the data movement applications and the network to effectively use the parallel Wide Area Network (WAN).

A useful tool called pmchart is available to monitor parallel data movement to or from SGI systems such as Edison.[7]

[7] If you have questions, problems regarding WAN file transfer, would like to monitor your transfer, or use pftp in a batch mode call Tom Pratt at 844-6725.

INDEX

Allocation and Access to Resources, 2

Batch
  Batch Job Examples - DPCS, 9
  Batch nodes, 2
Bugs
  General Bugs, 6
  Memory Bugs, 6

Compiler Options, 5
Compilers, 4
Compiling 64-bit Programs, 5

Data Services
  Pre & Post Processing Data Services, 14-16

Establishing cross-cell accounts, 2

File Movement, 17-18. *See also* Data Services

GPFS, 3, 9
Grid Services, 10-14
Group Account, 4

Ice, 1
Interactive
  Interactive job examples using POE, 8
  Interactive partitions, 3

Libraries, 5
Linker Options, 5
Logging into LLNL platforms, 3

Monitoring, 2, 11

MPI Performance, 6

Nemesis Tools, 15

Paths and environment, 4
pftp, 17-18
Processor Performance, 6-7

Remote Visualization, 15-17
Resource Status URLs, 2, 7, 11

SEACAS on White, 15
Shared Areas for Projects/Users, 4
Simulation Development Environment, 3-7
snow, 1-3, 6-9
Storage, 3. *See also* File Movement
  HPSS, 3, 14, 18

User Support Points of Contact, 1

VIEWS partition, 1, 3

White, 1-4, 7

WWW URLs:

  https://www-isn.sandia.gov/discom/services, 1
  https://secureweb.sandia.gov/discom/services, 1
  https://www.llnl.gov/lcforms/group.html, 4
  https://www.llnl.gov/lcforms/usr_gapps.html, 4
  http://www.kai.com/C_plus_plus/Current/doc, 4
  http://www.llnl.gov/asci/platforms/white/software, 4
  http://www.llnl.gov/asci/platforms/white/index.html, 7
  http://gridservices.sandia.gov/, 10
  http://sass2248.jal.sandia.gov/SEACAS/Documentation/SEACAS.html, 15