

The RWTH SunFire SMP-Cluster

User's Guide, Version 3.0

Errata and Addendum

January 2003

Dieter an Mey, Center for Computing and Communication, Aachen University
(Rechen- und Kommunikationszentrum der RWTH Aachen)

anmey@rz.rwth-aachen.de

Table of Contents

1 Introduction	2
2 Hardware.....	2
3 Operating System.....	3
3.1 Addressing Modes.....	3
3.2 Batch Job Administration.....	3
4 Programming/Tuning.....	3
4.1 Sun Compilers.....	3
5 Parallelization.....	5
5.1 Message passing with MPI.....	5
5.1.1 Sun MPI.....	5
5.1.1.1 Placing the MPI-Tasks with mprun.....	5
5.1.1.2 Input and output control with mprun.....	5
5.1.1.3 Handling MPI program runs.....	5
5.1.1.4 Sun MPI environment variables.....	5
5.1.2 mpich RWTH.....	5
6 Debugging.....	6
6.1 Static program analysis.....	6
6.2 Dynamic program analysis.....	6
6.3 Debuggers.....	6
6.3.1 dbx.....	6
6.3.2 Prism.....	6
6.3.3 TotalView.....	6
6.3.3.1 Invocation of TotalView for serial programs.....	6
6.3.3.2 Debugging of Sun-MPI programs RWTH.....	6

1 Introduction

- **The Forte Developer Version 7 compilers (which have been integrated into the Sun ONE Studio 7 Development Tools) are now in production mode and the default compilers. [NEW](#)**

The following Forte 7 Developer Version 7 Update 1 compilers can be tested after initialization with

```
. Forte71.init
```

- Sun HPC ClusterTools Version 4.0, Sun's MPI implementation and environment is still in production mode.

Version 5.0 is currently tested and will be put into production presumably in February 2003. MPI version 2 will be fully supported then. As these versions do not coexist, users cannot access version 5.0 so far. [NEW](#)

- **TotalView Version 6.0, Etnus' latest parallel debugger version is now in production mode and supports Sun's latest compilers and the debugging of programs using the 64 bit addressing mode. Debugging MPI programs will be improved with the upcoming HPC ClusterTools Version 5.0. [NEW](#)**

- VampirTrace Version 2.0 and Vampir Version 2.5, Pallas' tools for runtime analysis of MPI programs are still in production mode.

VampirTrace Version 3.0 and Vampir Version 3.0 are in test mode now, they will be accessible together with HPC ClusterTools Version 5.0 in February supporting 64 bit addressing mode. [NEW](#)

- KAP/Pro Toolset Version 4.0, KAI's OpenMP tools including
- the KCC C++ Compiler Version 4.0
- some details about the Solaris 9 operating system (It will successively be installed on the Sun Fire systems from 4Q 2002 on and particularly effects the runtime behaviour of the Sun Fire 15 K systems [RWTH](#))

2 Hardware

3 Operating System

3.1 Addressing Modes

3.2 Batch Job Administration

The most important **resource parameters** are (see **qsub**-parameter **-l**):

-l march=sf-15k -l march=sf-6800 CORRECTED	jobs can be directed to the Sun Fire 15K or to the Sun Fire 6800 nodes. Normally the use of these parameters is not recommended.
-l hw_counters NEW	Specify this resource, if you want to collect performance information (collect command)

4 Programming/Tuning

4.1 Sun Compilers

The Forte Developer Version 7 compilers (which have been integrated into the Sun ONE Studio 7 Development Tools) are now in production mode and the default compilers. You can choose the compiler version by modification of the search path with one of the following commands **RWTH**

```
$ . Forte61.init      # support will be discontinued
$ . Forte62.init
$ . Forte70.init      # now the default NEW
$ . Forte71.init      # in test mode NEW
```

We recommend that you always recompile your code with the latest production compiler for performance reasons and bug fixes.

Check the compiler version which you are currently using with the option

```
-v
```

or with the command

```
$ dumpstabs object_file
```

Online information in addition to the manual pages can be found by directing your browser to the local file

```
file:///opt/SUNWspro/docs/index.html
```

or to the website

```
http://docs.sun.com/db/coll/771.2
```

Particularly the new features are described in

<http://docs.sun.com/source/816-452/1.html>

NEW From version 7 on there no longer is a separate Fortran 77 compiler available. But there is an additional option in the new Fortran95 compiler improving the compatibility to Fortran 77

-f77

which has several suboptions. Using this option without any suboption list expands to

-fttrap=%none -f77=%all

which enables all compatibility features at the same time and also mimics the Fortran 77's behavior regarding arithmetic exception trapping. We recommend to add

-f77 -fttrap=common

in order to revert to the f95 trapping, which is considered to be safer.

When linking to old f77 object binaries, you may want to add the option

-xlang=f77

at the link step.

Detailed information about compatibility issues between Fortran 77 and Fortran 95 can be found in

http://docs.sun.com/source/816-2457/5_f77.html

5 Parallelization

5.1 Message passing with MPI

5.1.1 Sun MPI

5.1.1.1 Placing the MPI-Tasks with `mprun`

5.1.1.2 Input and output control with `mprun`

5.1.1.3 Handling MPI program runs

5.1.1.4 Sun MPI environment variables

Numerous environment variables can govern the behavior of an MPI program and improve its performance.

NEW The upcoming **HPC ClusterTools Version 5.0** includes a novel profiling tool **mpprof** which is easy to use and gives hints for setting additional environment variables which might improve the performance of a similar program run.

After enabling MPI profiling by setting the environment variable

```
$ export MPI_PROFILE=1
$ mprun -J -np n ... a.out
```

the MPI program run will write out profiling data for the MPI process ranks to a set of intermediate files, one file per process rank, as well as an index file pointing to the intermediate files. The **mpprof** command then generates a report of the performance characteristics of this MPI program

```
$ mpprof mpprof.index.cre.jid
```

with *jid* being the job ID of the cluster runtime environment (CRE) printed out with the `-J` option of the **mprun** command.

The collection of profiling data can be controlled by additional environment variables which are described in the manual page (**man mpprof**).

5.1.2 **mpich** **RWTH**

With the improved interoperability of the latest version 6.0 of the Totalview debugger and the upcoming HPC ClusterTools 5.0 we will no longer support **mpich** on our Sun platform. **NEW**

6 Debugging

6.1 Static program analysis

6.2 Dynamic program analysis

6.3 Debuggers

6.3.1 dbx

6.3.2 Prism

6.3.3 TotalView

The state-of-the-art debugger TotalView from Etnus (<http://www.etnus.com/>) can be used to debug serial and parallel Fortran, C and C++ programs. It is available on all major platforms.

NEW The latest version 6.0 of the **TotalView** debugger finally supports the latest Sun compilers.

It is recommended to no longer use the version 6 Update 1 compilers. Their support will be discontinued.

6.3.3.1 Invocation of TotalView for serial programs

```
$ . totalview.init
$ totalview program [corefile]
```

6.3.3.2 Debugging of Sun-MPI programs **RWTH**

At present a preliminary wrapper program has to be used, in order to debug a program compiled and linked with Sun MPI:

```
$ . totalview.init
$ totalview tvrun -a -np 2 -l "$(hostname) 2" a.out
```

NEW Once the HPC ClusterTools 5.0 will be in production, debugging MPI programs will become a bit more comfortable and message queues will be accessible by TotalView.

```
$ . totalview.init
$ totalview mprun -a -np 2 -l "$(hostname) 2" a.out
```