PGI® Server 9.0
PGI® Workstation 9.0
Installation Guide

Release 9.0

The Portland Group®
STMicroelectronics
Two Centerpointe Drive
Lake Oswego, OR 97035
Contents

1. Release 9.0 Introduction .............................................................................................................. 1
   Product Overview .......................................................................................................................... 1
   Terms and Definitions .................................................................................................................... 2
   Supported Processors ................................................................................................................... 2
   Supported Operating Systems ....................................................................................................... 3
   Product Support ............................................................................................................................ 6

2. PGI Release 9.0 Installation Overview ......................................................................................... 7
   Before You Begin .......................................................................................................................... 7

3. PGI Release 9.0 Licensing ........................................................................................................... 9
   Permanent and Trial License Keys ............................................................................................... 9
   Trial Licensing Key Constraints ................................................................................................. 10
   License Keys and System Configurations .................................................................................... 10
   PGI Workstation Licensing ......................................................................................................... 10
   PGI Server Licensing ................................................................................................................... 10
   The FLEXnet License Manager .................................................................................................. 10
   License Support .......................................................................................................................... 11

4. Installations on Linux ................................................................................................................ 13
   Prepare to Install on Linux .......................................................................................................... 13
   Installation Steps for Linux ........................................................................................................... 15
   Typical Directory Structure for Linux ......................................................................................... 23
   End-user Environment Settings on Linux .................................................................................... 24
   Common Linux Installation Issues ............................................................................................... 25
   Java Runtime Environment (JRE) .............................................................................................. 26

5. Installations on Windows, SFU, and SUA .................................................................................. 27
   Preparing to Install on Windows, SFU, SUA ............................................................................. 27
   Installing Required Utilities for SUA ......................................................................................... 28
   Installing Required Utilities for SFU ........................................................................................... 28
   Installation Steps for Windows, SFU, SUA .................................................................................. 31
Figures

4.1. Linux Installation Overview ................................................................. 14
5.1. PGI Installation on Windows Overview .................................................. 30
6.1. PGI Installation on Mac OS X Overview ............................................... 45

Tables

1.1. Processors Supported by PGI 9.0 ............................................................... 3
1.2. Operating Systems and Features Supported in PGI 9.0 .......................... 4
4.1. Linux Directory Structure Sample ......................................................... 23
4.2. Linux Directory Structure Sample ......................................................... 24
5.1. Default Installation Directories by Platform ......................................... 36
5.2. Win64 Sample Directory Structure ...................................................... 37
5.3. Win32 Sample Directory Structure ...................................................... 37
5.4. Sample SFU Directory Structure ......................................................... 38
5.5. Sample SUA32 Directory Structure .................................................... 38
5.6. Sample SUA32 Directory Structure .................................................... 38
6.1. Sample Mac OS Directory Structure .................................................... 49
Chapter 1. Release 9.0 Introduction

Welcome to Release 9.0 of PGI Workstation and PGI Server, a set of Fortran, C, and C++ compilers and development tools for 32-bit and 64-bit x86-compatible processor-based workstations and servers running versions of the Linux, Windows, and Mac OS operating systems.

All workstation-class compilers and tools products from The Portland Group (PGI Fortran Workstation, for example) are subsets of the PGI Workstation Complete product. These workstation-class products provide a node-locked single-user license, meaning one user at a time can compile on the one system on which the PGI Workstation compilers and tools are installed.

PGI Server products are offered in configurations identical to the workstation-class products, but provide network-floating multi-user licenses. This means that two or more users can use the PGI compilers and tools concurrently on any compatible system networked to the system on which the PGI Server compilers are installed.

This installation information applies to all workstation-class and server-class compiler products from The Portland Group.

Product Overview

Release 9.0 of PGI Workstation and PGI Server the PGI CDK includes the following components:

- PGF95 OpenMP* and auto-parallelizing Fortran 90/95 compiler.
- PGF77 OpenMP and auto-parallelizing FORTRAN 77 compiler.
- PGHPF data parallel High Performance Fortran compiler.

Note

PGHPF is supported only on Linux platforms.

- PGCC OpenMP and auto-parallelizing ANSI C99 and K&R C compiler.
- PGC++ OpenMP and auto-parallelizing ANSI C++ compiler.
- PGPROF graphical MPI/OpenMP/multi-thread performance profiler.
- PGDBG graphical MPI/OpenMP/multi-thread symbolic debugger.
• MPICH MPI libraries, version 1.2.7, for both 32-bit and 64-bit development environments (Linux only).
• Online documentation in PDF, HTML and man page formats.
• A UNIX-like shell environment for Win32 and Win64 platforms.

Depending on the product configuration you purchased, you may not have licensed all of the above components.

The MPI profiler and debugger included with PGI Workstation are limited to processes on a single node. PGI Workstation can be installed on a single computer, and that computer can be used to develop, debug, and profile MPI applications. The PGI CDK Cluster Development Kit supports general development on clusters.

Terms and Definitions

This Installation Guide contains a number of terms and definitions with which you may or may not be familiar. If you encounter a term in these notes with which you are not familiar, please refer to the online glossary at www.pgroup.com/support/definitions.htm

These two terms are used throughout the documentation to reflect groups of processors:

• **AMD64** – a 64-bit processor from AMD designed to be binary compatible with 32-bit x86 processors, and incorporating new features such as additional registers and 64-bit addressing support for improved performance and greatly increased memory range. This term includes the AMD™ Athlon64™, AMD Opteron™, AMD Turion™, AMD Barcelona, AMD Shanghai, and AMD Istanbul processors.

• **Intel 64** – a 64-bit IA32 processor with Extended Memory 64-bit Technology extensions designed to be binary compatible with AMD64 processors. This includes Intel Pentium 4, Intel Xeon, Intel Core 2, Intel Penryn, and Intel Core i7 (Nehalem) processors.

**Supported Processors**

Table 1.1 lists the processors on which Release 9.0 of the PGI compilers and tools is supported. The table also includes the CPUs available and supported in multi-core versions.

The `-tp <target>` command-line option generates executables that utilize features and optimizations specific to a given CPU and operating system environment. Compilers included in a 64-bit/32-bit PGI installation can produce executables targeted to any 64-bit or 32-bit target, including cross-targeting for AMD and Intel 64-bit AMD64 compatible CPUs.

In addition to the capability to generate binaries optimized for specific AMD or Intel processors, the PGI 9.0 compilers can produce PGI Unified Binary object or executable files containing code streams fully optimized and supported for both AMD and Intel x64 CPUs. To produce unified binary files, you use one of the following `-tp` command-line options: `-tp x64` or `-tp <target1>,<target2>,<target3>...`, where `<target>` is any of the valid values in Table 1.1.
Table 1.1. Processors Supported by PGI 9.0

<table>
<thead>
<tr>
<th>Brand</th>
<th>CPU</th>
<th>Target Processor -tp &lt;target&gt; [,.target...]</th>
<th>32-bit</th>
<th>64-bit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AMD</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Opteron Six-core Istanbul</td>
<td>istanbul-32</td>
<td></td>
<td>istanbul-64</td>
</tr>
<tr>
<td></td>
<td>Opteron Quad-core Shanghai</td>
<td>shanghai-32</td>
<td></td>
<td>shanghai-64</td>
</tr>
<tr>
<td></td>
<td>Opteron Quad-core Barcelona</td>
<td>barcelona-32</td>
<td></td>
<td>barcelona-64</td>
</tr>
<tr>
<td></td>
<td>Opteron Quad-core</td>
<td>k8-32</td>
<td>k8-32</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Opteron Rev E?F</td>
<td>k8-32</td>
<td>k8-32e</td>
<td>k8-64e</td>
</tr>
<tr>
<td></td>
<td>Turion / Turion 64</td>
<td>k8-32</td>
<td>k8-32e</td>
<td>k8-64e</td>
</tr>
<tr>
<td></td>
<td>Athlon64</td>
<td>NA</td>
<td></td>
<td>k8-64e</td>
</tr>
<tr>
<td></td>
<td>Athlon</td>
<td>athlon</td>
<td></td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td>Athlon XP/MP</td>
<td>athlonxp</td>
<td></td>
<td>NA</td>
</tr>
<tr>
<td><strong>Intel</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Core i7 - Nehalem</td>
<td>nehalem-32</td>
<td></td>
<td>nehalem-64</td>
</tr>
<tr>
<td></td>
<td>Penryn</td>
<td>penryn-32</td>
<td></td>
<td>penryn-64</td>
</tr>
<tr>
<td></td>
<td>Core 2</td>
<td>core2-32</td>
<td></td>
<td>core2-64</td>
</tr>
<tr>
<td></td>
<td>P4/Xeon EM64T</td>
<td>p7-32</td>
<td>p7-64</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Xeon Pentium4</td>
<td>p7-32</td>
<td>NA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pentium III</td>
<td>piii</td>
<td>NA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pentium II</td>
<td>p6</td>
<td>NA</td>
<td></td>
</tr>
<tr>
<td><strong>Generic</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Generic x86</td>
<td>p5 or px-32</td>
<td></td>
<td>NA</td>
</tr>
</tbody>
</table>

**Supported Operating Systems**

Table 1.2 lists the operating systems, and their equivalents, on which Release 9.0 of the PGI compilers and tools is supported.

To determine if Release 9.0 will install and run under a Linux equivalent version, such as Mandrake®, Debian®, Gentoo®, and so on, check the table for a supported system with the same glibc and gcc versions. Version differences in other operating system components can cause difficulties, but often these can be overcome with minor adjustments to the PGI software installation or operating system environment.

- Newer distributions of the Linux operating systems include support for x64 compatible processors and are designated 64-bit in the table. These are the only distributions on which the 64-bit versions of the PGI compilers and tools will fully install.
• If you attempt to install the 64-bit/32-bit Linux version on a system running a 32-bit Linux distribution, only the 32-bit PGI compilers and tools are installed.

• If you attempt to install the 64-bit Windows version on a system running 32-bit Windows, the installation fails.

Most newer Linux distributions support the *Native Posix Threads Library (NPTL)*, a new threads library that can be utilized in place of the *libpthreads* library available in earlier versions of Linux. Distributions that include NPTL are designated in the table. Parallel executables generated using the *OpenMP* and auto-parallelization features of the PGI compilers will automatically make use of NPTL on distributions when it is available. In addition, the PGDBG debugger is capable of debugging executables built using either NPTL or earlier thread library implementations.

Multi-socket AMD Opteron processor-based servers use a *NUMA* (Non-Uniform Memory Access) architecture in which the memory latency from a given processor to a given portion of memory can vary. Newer Linux distributions, including SuSE 9/10 and SLES 9/10, include NUMA libraries that can be leveraged by a compiler and associated runtime libraries to optimize placement of data in memory.

In the table headings:

**HT** = hyper-threading

**NPTL** = Native POSIX Threads Library

**NUMA** = Non-Uniform Memory Access

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Type</th>
<th>64-bit</th>
<th>HT</th>
<th>pgC++</th>
<th>pgdb</th>
<th>NPTL</th>
<th>NUMA</th>
<th>glibc</th>
<th>GCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL 5.3</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.5</td>
<td>4.1.2</td>
<td></td>
</tr>
<tr>
<td>RHEL 5.0</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.5</td>
<td>4.1.2</td>
<td></td>
</tr>
<tr>
<td>RHEL 4.0</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.3.4</td>
<td>3.4.3</td>
<td></td>
</tr>
<tr>
<td>RHEL 3.0</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.3.2</td>
<td>3.2.3</td>
<td></td>
</tr>
<tr>
<td>Fedora 11</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.9</td>
<td>4.3.3</td>
<td></td>
</tr>
<tr>
<td>Fedora 10</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.9</td>
<td>4.3.2</td>
<td></td>
</tr>
<tr>
<td>Fedora 9</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.8</td>
<td>4.3.0</td>
<td></td>
</tr>
<tr>
<td>Fedora 8</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.7</td>
<td>4.1.2</td>
<td></td>
</tr>
<tr>
<td>Fedora 7</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.6</td>
<td>4.1.2</td>
<td></td>
</tr>
<tr>
<td>Fedora 6</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.5</td>
<td>4.1.1</td>
<td></td>
</tr>
<tr>
<td>Fedora 5</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.4</td>
<td>4.1.0</td>
<td></td>
</tr>
<tr>
<td>Fedora 4</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.3.5</td>
<td>4.0.0</td>
<td></td>
</tr>
<tr>
<td>Fedora 3</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.3.3</td>
<td>3.4.2</td>
<td></td>
</tr>
<tr>
<td>Fedora 2</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.3.3</td>
<td>3.3.3</td>
<td></td>
</tr>
<tr>
<td>SuSE 11.1</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.9</td>
<td>4.3.3</td>
<td></td>
</tr>
<tr>
<td>SuSE 11.0</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.8</td>
<td>4.3.1</td>
<td></td>
</tr>
</tbody>
</table>
### Distribution Specifications

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Type</th>
<th>64-bit</th>
<th>HT</th>
<th>pgC++</th>
<th>pgdbg</th>
<th>NPTL</th>
<th>NUMA</th>
<th>glibc</th>
<th>GCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>SuSE 10.3</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.6.1</td>
</tr>
<tr>
<td>SuSE 10.2</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.5</td>
</tr>
<tr>
<td>SuSE 10.1</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.4</td>
</tr>
<tr>
<td>SuSE 10.0</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.3.5</td>
</tr>
<tr>
<td>SuSE 9.3</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.3.4</td>
</tr>
<tr>
<td>SuSE 9.2</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.3.3</td>
</tr>
<tr>
<td>SuSE 9.1</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.3.3</td>
</tr>
<tr>
<td>SuSE 9.0</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>2.3.2</td>
</tr>
<tr>
<td>SLES 11</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.9</td>
</tr>
<tr>
<td>SLES 10</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.4</td>
</tr>
<tr>
<td>SLES 9</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>2.3.3</td>
</tr>
<tr>
<td>RedHat 9.0</td>
<td>Linux</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>2.3.2</td>
</tr>
<tr>
<td>Ubuntu 9.04</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.9</td>
</tr>
<tr>
<td>Ubuntu 8.10</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.8</td>
</tr>
<tr>
<td>Ubuntu 8.04</td>
<td>Linux</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>2.7</td>
</tr>
</tbody>
</table>

**Microsoft Windows**

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Type</th>
<th>64-bit</th>
<th>HT</th>
<th>pgC++</th>
<th>pgdbg</th>
<th>NPTL</th>
<th>NUMA</th>
<th>glibc</th>
<th>GCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>XP</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>XP x64</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>2003</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>2003 x64</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>2008</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>2008 x64</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>SFU</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>SFU</td>
<td>3.3</td>
<td></td>
</tr>
<tr>
<td>SUA</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>SUA</td>
<td>3.3</td>
<td></td>
</tr>
<tr>
<td>SUA x64</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>SUA</td>
<td>3.3</td>
<td></td>
</tr>
<tr>
<td>Vista</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Vista x64</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>HPC Server 2008</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>

**Apple**

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Type</th>
<th>64-bit</th>
<th>HT</th>
<th>pgC++</th>
<th>pgdbg</th>
<th>NPTL</th>
<th>NUMA</th>
<th>glibc</th>
<th>GCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tiger</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>4.01</td>
</tr>
<tr>
<td>Leopard</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>4.01</td>
</tr>
</tbody>
</table>

---

**Note**

http://www.pgroup.com/support/install.htm lists any new operating system distributions that may be explicitly supported by the PGI compilers. If your operating system is newer than any of those listed in Table 1.2, the installation may still be successful.
Product Support

All new PGI licenses include 60 days of PGI Subscription Service.

The PGI Subscription Service provides support and other benefits, including:

- Ongoing technical support.

  Support requests may be sent in a number of ways:
  
  - By electronic mail to trs@pgroup.com
  - Faxed to +1-503-682-2637
  - By using the online support request form.

  Phone support is not currently available.

- Notification by email when maintenance releases occur and are available for electronic download and installation.

- Release upgrades for licensed Product(s) at no additional cost, except for any administrative fee that may apply.

- Full license fee credits on Product upgrades, except for any administrative fee that may apply. "Product upgrades" refer to exchanging one Product license for a more expensive Product license, and is not the same as a version or Release upgrade previously referenced.

- Full license fee credits on user-count upgrades, except for any administrative fee that may apply.

  **Important**

  To continue receiving these benefits after 60 days, you can purchase an extension to your PGI Service Subscription. Extensions are available in yearly increments.

  Contact us at sales@pgroup.com if you would like information regarding the subscription service for the PGI products you have purchased.
Chapter 2. PGI Release 9.0
Installation Overview

This section provides an overview of the steps required to successfully install PGI Workstation or PGI Server. The following chapters provide the details of each of the steps. Specifically, Chapter 3 describes licensing generally. Chapter 4 describes how to install PGI Workstation or PGI Server in a generic manner on Linux, including how to install and run a FLEXnet license daemon on Linux. Chapter 5 describes how to install on a Windows system, and Chapter 6 describes how to install on an Apple Mac OS system.

Before You Begin

Before you begin the installation, it is essential that you understand the flow of the installation process. There are three stages of the process:

- Prepare to install – verifying that you have all the required information and software.
- Install the software – installing the software appropriate for your operating system
- Generation of license keys using the PGI website, installation of the license keys, and starting the license server.

The following illustration provides a high-level overview of the installation process.
For more complete information on these steps and the specific actions to take for your operating system, refer to the following chapters.
Chapter 3. PGI Release 9.0
Licensing

The PGI compilers and tools are license-managed.

Before discussing licensing, it is useful to have common terminology. These two terms are often confused, so they are clarified here:

- **License** - a legal agreement between ST and PGI end-users, to which users assent upon installation of any PGI product. The terms of the License are kept up-to-date in documents on pgroup.com and in the `$PGI/<platform>/<rel_number>` directory of every PGI SW installation.

- **License keys** - ASCII text strings that enable use of the PGI software and are intended to enforce the terms of the License. License keys are generated by each PGI end-user on pgroup.com using a unique hostid and are typically stored in a file called `license.dat` that is accessible to the systems for which the PGI software is licensed at a given site.

There are two types of license keys: permanent and trial.

**Permanent and Trial License Keys**

PGI Visual Fortran includes the PGI License Setup tool to help automate your license retrieval and installation process. You use this tool to obtain either a trial or a permanent license key.

- When you purchase a **permanent** PGI license, the email order confirmation you receive includes complete instructions for logging on to the pgroup.com web page and generating permanent license keys.

- When you register for a **trial** license, you generate trial keys using the web page: `www.pgroup.com/login.php`.

For more detailed information on how to do obtain a license key, refer to Step 5 in “Installation Steps for Linux,” on page 15.

**Note**

You must install the PGI software before you obtain your license keys because the license key generation process requires information that is generated during the software installation.
Trial Licensing Key Constraints

**Note**

At the conclusion of the trial period, the PGI compilers and tools and any executable files generated prior to the installation of permanent license keys will cease to function.

Any executables, object files, or libraries created using the PGI compilers with a trial key must be recompiled with permanent license keys in place.

License Keys and System Configurations

Executable files generated with permanent license keys in place are unconstrained, and will run on any compatible system regardless of whether the PGI compilers are installed.

**Important**

If you change the configuration of your system by adding or removing hardware, your license key may become invalid. Please contact The Portland Group if you expect to reconfigure your system to ensure that you do not temporarily lose the use of the PGI compilers and tools.

**PGI Workstation Licensing**

PGI Workstation is node-locked to a single system in a way that allows any user of the system to use the PGI compilers and tools. However, only a single user is allowed to run any of the PGI compilers or tools at a given time.

**PGI Server Licensing**

PGI Server supports multi-user, network floating licenses. Multiple users can use the PGI compilers and tools simultaneously from multiple systems on a network when those systems have a properly configured version of PGI Server installed. The number of seats purchased for the license determines the limitation on the number of concurrent users.

On Linux, PGI Server may be installed locally on each machine on a network or may be installed once on a shared file system available to each machine. If you select the second method, a network install, adding another machine to the group running the compilers is a much simpler process in this release; you adjust to the unique characteristics of the newly added system with a customization script that must be executed on each machine in the group.

**Note**

On Windows and Mac OS X, PGI Server must be installed locally on each machine.

**The FLEXnet License Manager**

PGI CDKPGI Workstation and PGI Server software licensing is managed using the FLEXnet license management system from Macrovision. As part of the process of installing the PGI compilers and tools, you install and configure the FLEXnet license management software. The instructions in *Chapter 4, “Installations on Linux”*
describe how to configure license daemons for Linux, Windows or Mac OS, including installation and start-up of the license daemon, and proper initialization of the LM_LICENSE_FILE and FLEXLM_BATCH environment variables.

License Support

All new PGI licenses include 60 days of PGI Subscription Service. For more information about this service and how to extend it, refer to “Product Support,” on page 6.
Chapter 4. Installations on Linux

This chapter describes how to install PGI Workstation or PGI Server on a Linux system. It covers local and network installs and is applicable to permanent or trial installations.

- For installations on 32-bit x86 systems, the PGI installation script installs only the linux86 versions of the PGI compilers and tools.
- For installations on 64-bit x64 systems running a linux86-64 execution and development environment, the PGI installation script installs the linux86-64 version of the PGI compilers and tools.
- If the 32-bit gcc development package is already installed on the system, the 32-bit linux86 tools are also installed on a 64-bit x64 system.

Prepare to Install on Linux

In preparation for installing PGI Workstation or PGI Server on Linux, first review the overview of the Linux installation process illustrated in Figure 4.1.

Note

For Linux installations, each user must set their environment variables properly to access the software, as described in section 4.4, End-user Environment Settings on Linux.

The 32-bit and 64-bit compilers, tools, and supporting components have the same command names. Further, the environment you target by default, linux86-64 or linux86, depends on the version of the compiler that comes first in your path settings.

In a traditional local installation, the default installation base directory is /opt/pgi.

Network Installations:

If you choose to perform a network install, you should specify:

- A shared file system for the installation base directory.
- A second directory name that is local to each of the systems where the PGI compilers and tools are used. This local directory will contain the libraries to use when compiling and running on that machine.
This directory selection approach allows a network installation to support a network of machines that run different versions of Linux.

Figure 4.1. Linux Installation Overview

- Select which products
- Select 32- or 64-bit
- Determine local or on network install
- Check disk space availability
- Start Shell Command Window
- Download from web site
- Unpack the file
- Do I have write permissions?
- Can I access the software?
- Run the install script.
- Consent to the PGI end-user licence agreement (EULA).
- Execute appropriate commands to make products available and to prepare environment for licensing
- Use the -V option on any compiler command to verify the release number
- Have your PGI username, password, FLEXlm hostid, and computer hostname
- Obtain licenses from web site
- Place licenses in license.dat file
- Issue the lmgrd.rc start command to start the license server and daemon

HTML and PDF versions are available
- Did I bookmark the location?

For Network Installations Only:
- Run local installation script on each system on network where compilers and tools are to be available
To prepare for the installation:

- Locate your PGI Order Confirmation email.
  
  This email contains instructions on how to download the PGI software from the PGI web page as well as other information you may need for generating your permanent license. It also contains your PGI Product Identification Number (PIN) that you need should you need to contact PGI.

- Bring up a shell command window on your system.
  
  The installation instructions assume you are using csh, sh, ksh, bash, or some compatible shell. If you are using a shell that is not compatible with one of these shells, appropriate modifications are necessary when setting environment variables.

- Verify you have enough free disk space.
  
  - linux86 platform: PGI installation requires 250 MB of free disk space.
  
  - linux86-64 platform: PGI installation requires 1.4 GB of free disk space.

**Installation Steps for Linux**

Follow these instructions to install the software:

1. Unpack the PGI Software
   
   Once you download the software from http://www.pgroup.com or another electronic distribution site, then in the instructions that follow, replace `<tarfile>` with the name of the file that was downloaded.

   **Note**

   The PGI products cannot be installed into the same directory where the tar file is unpacked.

   Use the following command sequence to unpack the tar file in a temporary directory before installation:

   ```
   % mkdir /tmp/pgi
   % mv <tarfile>.tar.gz /tmp/pgi
   % cd /tmp/pgi
   % tar xpfz <tarfile>.tar.gz
   ```

2. Run the Install Script.

   The `install` script *must* be run to properly install the software.

   Execute the following script in the directory where you unpacked the tar file:

   ```
   % ./install
   ```

   When you run this script, you must:

   - Consent to the PGI end-user license agreement (EULA)
   - Determine whether to perform a traditional local installation or a network installation
   - Determine whether to install the optional ACML math library from AMD
• Define where to place the installation directory.

After the software is installed, the install script performs system-specific customization and then initializes the licensing.

Note

For a network installation, you are asked for a common local directory. This local directory will be created once on each system utilizing the network installation; further, it must be created on each system before adding that system to the network using the compilers.

Note

If you have difficulty running this script, especially on a Slackware Linux system, check the permissions on /dev/null. Permissions should be set to “crw-rw-rw-”. If necessary, reset permissions to this value; to do this, super-user permissions are required.

At the end of the installation, the install script prints a message similar to this:

To obtain an evaluation license, go to: www.pgroup.com/login and use your web-user access codes (email address and password) and the information below to generate a trial license.

For a permanent license, please read the order confirmation that you received. Connect to www.pgroup.com/login with the username and password provided in the order confirmation.

FLEXlm hostid: <your host id>
Hostname: <your host name>
Installation: /opt/pgi
PGI Release: 9.0-2

You need the FLEXlm hostid and Hostname in Step 3 below.

Tip

In case you need to retrieve FLEXlm information at a later time, the preceding message is saved to the file /opt/pgi/license.info, where /opt/pgi is the installation directory.


When the install script has completed, execute the following commands to make the PGI products accessible and to initialize your environment for use of FLEXnet.

For linux86-64:

To use the linux86-64 version of the compilers and tools, execute the following commands, assuming you have installed in the default /opt/pgi directory.

In csh, use these commands:
% setenv PGI /opt/pgi
% set path=(/opt/pgi/linux86-64/9.0/bin $path)
% setenv MANPATH "$MANPATH":/opt/pgi/linux86-64/9.0/man
% setenv LM_LICENSE_FILE "$LM_LICENSE_FILE":/opt/pgi/license.dat

In bash, sh, or ksh, use these commands:

$ PGI=/opt/pgi; export PGI
$ PATH=/opt/pgi/linux86-64/9.0/bin:$PATH; export PATH
$ MANPATH=$MANPATH:/opt/pgi/linux86-64/9.0/man; export MANPATH
$ LM_LICENSE_FILE= $LM_LICENSE_FILE:/opt/pgi/license.dat
$ export LM_LICENSE_FILE

For \texttt{linux86}:

To use only the \texttt{linux86} version of the compilers and tools, or to target \texttt{linux86} as the default, use a setup similar to the previous one, changing the path settings as illustrated in the following commands.

In csh, use these commands:

% setenv PGI /opt/pgi
% set path=(/opt/pgi/linux86/9.0/bin $path)
% setenv MANPATH "$MANPATH":/opt/pgi/linux86/9.0/man
% setenv LM_LICENSE_FILE "$LM_LICENSE_FILE":/opt/pgi/license.dat

In bash, sh, or ksh, use these commands:

$ PATH=/opt/pgi/linux86/9.0/bin:$PATH; export PATH
$ MANPATH=$MANPATH:/opt/pgi/linux86/9.0/man; export MANPATH
$ LM_LICENSE_FILE= $LM_LICENSE_FILE:/opt/pgi/license.dat;
$ export LM_LICENSE_FILE

\textbf{Note}

You should add these commands to your shell startup files to ensure that you have access to the PGI products upon future logins.

4. **Verify the Release Number of the Installed Software.**

To verify the release number of the products you have installed, use the \texttt{-V} option on any of the compiler commands, as illustrated in the following examples. If you use \texttt{--v} instead, you also see the sequence of steps the compiler uses to compile and link programs for execution on your system.

- For Fortran 77, use: \texttt{pgf77 \textasciitilde V x.f}
- For Fortran 95, use: \texttt{pgf95 \textasciitilde V x.f}
- For HPF, use: \texttt{pghpf \textasciitilde V x.f}
- For C++, use: \texttt{pgCC \textasciitilde V x.c}
- For ANSI C, use: \texttt{pgcc \textasciitilde V x.c}
Note

These commands can be successfully executed even if the files x.f or x.c do not exist and you have not completed the licensing phase of the installation. The PGI compiler drivers support this as a quick check that you have installed the proper version of the compilers and have initialized your environment to enable access to that version.

5. Generate and Install License Keys.

All of the PGI compilers and tools are license-managed and require installation of license keys to make the PGI software operational. The other components of the PGI CDK, including MPICH, MPICH2, MVAPICH, ScaLAPACK, and the OpenPBS batch scheduler, are open source products that are not license-managed.

To obtain license keys, you need the following information:

- An account on the PGI website. You probably created this account when you downloaded the PGI software.
- If you purchased a license without creating an account, one was created for you when your order was processed. Please check for an activation email from webmaster@pgroup.com.
- If you don’t have an account, you can create one at: https://www.pgroup.com/account/register.php
- The FLEXlm hostid and hostname of the computer on which the software is installed, which is echoed to your screen by the installer.

Note

You can also obtain your FLEXnet hostid by using the following command after you have installed the products and initialized the environment variables:

```
% lmutil lmhostid
```

You see a message similar to the following one, in which 12345678abcd is unique to your machine.

The FLEXnet host ID of this machine is "12345678abcd".

Generate License Keys

Once you have logged on at www.pgroup.com/login using your PGI web-user email and password, you will see a screen similar to the following:
To generate permanent license keys:

1. Click Create permanent keys.
2. Click the PIN associated with the product for which you wish to generate license keys. If you don't see any PINs listed, you first need to tie one to your account.
   a. Obtain your PIN code from your original PGI order confirmation email.
   b. Click the link: Tie a PIN to this account.
   c. Follow the instructions provided.
3. Click License keys to generate the keys.

To generate trial license keys:

1. Click Create trial keys.
2. Accept the terms of the agreement.
3. Enter the hostid exactly as it appears in the message that is displayed during installation or when you issue the command:

   ```
   % lmutil lmhostid
   ```
4. Click the Generate license key button.

Install License Keys

Once you have generated your trial or permanent license keys, copy, and then paste them into the file: /opt/pgi/license.dat, substituting the appropriate installation directory path if you have not installed in the default /opt/pgi directory.

For example, if you have purchased PGI Fortran Workstation for Linux, the license.dat file should look similar to the following:
SERVER <hostname> <hostid> 27000
DAEMON pgroup
PACKAGE pgi90-workstation-fortran \ 
pgroupd 9.0 2A64FB18DEBE COMPONENTS= \ "pgi-f95-lin64:9.0 pgi-f77-lin64:9.0 \ pgi-hpf-lin64:9.0 pgi-cc-lin64:9.0 \ pgi-cpp-lin64:9.0 \ pgi-f95-lin32:9.0 pgi-f77-lin32:9.0 \ pgi-hpf-lin32:9.0 pgi-cc-lin32:9.0 \ pgi-cpp-lin32:9.0 \ pgi-prof:9.0 pgi-dbg:9.0 pgi-dbg-gui:9.0" \ OPTIONS=SUITE_RESERVED SIGN= \ "0D91 C4A7 3569 5A7C 65EE 91EC A9AA 706C \ 5EB8 C207 2DE8 9086 3926 4F84 BAF4 1C9D \ EC65 00BF 51A2 9A8D DF53 C460 7C10 663F \ 9C41 92B9 F9B2 6150 D6C1 A94"

FEATURE pgi-workstation-fortran pgroupd 9.0 \ 
31-dec-0 5 8B15A3507BE4 \ 
VENDOR_STRING=107209:8:ws DUP_GROUP=U \ 
SUITE_DUP_GROUP=U BORROW=336 TS_OK SIG2" 
"0501 9540 017D 9D8C CE97 E26B F967 A848 \ 864E 6EC2 7205 E535 3E6B B6B6 7ED3 0F83 \ 8FDC 48FE 3AD5 E6D0 5B2E 26E5 D18C 2E6B \ BC30 1838 1526 C3BF F081 D19F"

In your license file:

- `<hostid>` should match the hostid you submitted above when you generated your license keys.

- If necessary, you can enter or edit the `<hostname>` entry manually, but you cannot edit the `<hostid>` entry or you will invalidate the license keys.

- You have a similar unique Product ID Number (PIN) for your installation.

  The six digits immediately following the `=` in the feature line component, `107209` of
  `VENDOR_STRING=107209:8`, in this example, represent the PIN for this installation.

  **Note**

  Please include your PIN when contacting us regarding technical support for the products you have purchased. This PIN is also on your order confirmation email.

6. Start the License Manager daemon.

  **Important**

  If you are evaluating PGI software with trial keys, you do not need to perform this step and can proceed to Step 7.

**Installations in a directory other than the default** `/opt/pgi`
Note

The following refers to the shell script template for linux86-64. If you have installed only linux86, please substitute linux86 for linux86-64.

If you installed the compilers in a directory other than /opt/pgi, do this:

1. Edit the shell script template $PGI/linux86-64/9.0/bin/lmgrd.rc.
2. Substitute the correct installation directory for /opt/pgi in the section of the script entitled Where to find the PGI Software.
3. Save the file and exit the editor.

Issue the following command to start the license server and pgroupd license daemon running on your system:

```
% lmgrd.rc start
```

If you wish to stop the license server and license daemon at a later time, you can do so with the command:

```
% lmgrd.rc stop
```

Start license server upon reboot:

To start the license server and pgroupd daemon each time your system is booted:

1. Log in as root.

   Note

   You must be logged in as root to successfully execute these commands.

2. Verify you have set the PGI environment variable as described in Step 3 of this installation process.
3. Execute the following two commands:

   ```
   % cp $PGI/linux86/9.0/bin/lmgrd.rc /etc/init.d/lmgrd
   % ln -s /etc/init.d/lmgrd /etc/rc.d/rc3.d/S90lmgrd
   ```

There are two values in this example that may be different on your system:

- Your system’s default runlevel may be something other than ‘3’, the level used in this example. You can run /sbin/runlevel to check the system’s runlevel. If the runlevel on your systems is different, then you must set the correct subdirectory; use your system’s runlevel in place of the “3” in the preceding example.

- Your rc files may be in a directory other than the one in the example: /etc/init.d. If the rc files are in a directory such as /etc/rc.d/init.d, then substitute that location in the example.
chkconfig(8) Utility

Most Linux distributions include the chkconfig(8) utility that manages the runlevel scripts. If your system has this tool and you wish to use it, then run the following commands:

```bash
% cp $PGI/linux86/9.0/bin/lmgrd.rc /etc/rc.d/init.d/
% chkconfig --add lmgrd.rc
```

These commands create the appropriate links in the `/etc/rc.d` directory hierarchy. For more information on `chkconfig`, please refer to the manual page.

Important

Release 7.2 and higher included a newer version of the Macrovision FLEXnet software. The updated `lmgrd` and `pgroupd` daemons must be used in preference to versions shipped with previous releases of the PGI products.

You can co-install Release 9.0 with Release 8.x, 7.x, 6.x and/or 5.2; and you can use any of these versions of the compilers and tools with the new versions of `lmgrd` and `pgroupd` and a single Release 9.0 license file.

If you use this file to start `lmgrd` automatically after a reboot of your system, you must modify your `lmgrd.rc` file in the `/etc/rc.d` or `/etc/init.d` directory to use the new `lmgrd`.

For example, your `lmgrd.rc` file may look like this one, where `<target>` is replaced appropriately with `linux86` or `linux86-64`.

```bash
## Path to master daemon lmgrd
#LMGRD=$PGI/<target>/5.2/bin/lmgrd
LMGRD=$PGI/<target>/9.0/bin/lmgrd
## Command to stop lmgrd
#LMUTIL=$PGI/<target>/5.2/bin/lmutil
LMUTIL=$PGI/<target>/9.0/bin/lmutil
```

7. Review Documentation.

You can view the online HTML and PDF documentation using any web browser by opening the file:

`$PGI/linux86-64/9.0/doc/index.htm`

or

`$PGI/linux86/9.0/doc/index.htm`

You may want to bookmark this location for easy future reference to the online manuals.


Note

Skip this step if you are not using a network installation.
For a network installation, you must run the local installation script on each system on the network where the compilers and tools will be available for use.

If your installation base directory is /opt/pgi and the common local directory is /usr/pgi/shared/9.0, run the following commands on each system on the network.

```
/opt/pgi/linux86/9.0-2/bin/makelocalrc \
-x /opt/pgi/linux86/9.0-2 \
-net /usr/pgi/shared/9.0/opt/pgi/linux86-64/9.0-2/bin/makelocalrc \
-x /opt/pgi/linux86-64/9.0-2 \
-net /usr/pgi/shared/9.0
```

These commands create a system-dependent file `localrc.machinename` in both the `/opt/pgi/linux86/9.0-2/bin` directory and in `/opt/pgi/linux86-64/9.0-2/bin`. The commands also create the following three directories containing libraries and shared objects specific to the operating system and system libraries on that machine: `/usr/pgi/shared/9.0/lib`, `/usr/pgi/shared/9.0/liblf`, and `/usr/pgi/shared/9.0/lib64`.

**Note**

The `makelocalrc` command does allow the flexibility to have local directories that have different names on different machines. However, using the same directory on different machines allows users to easily move executables that use PGI-supplied shared libraries between systems.

Installation of the PGI products for Linux is now complete. For assistance with difficulties related to the installation, send email to trs@pgroup.com.

The following two sections contain information detailing the directory structure of the PGI installation, and instructions for how PGI end-users initialize environment and path settings to use the PGI compilers and tools.

**Typical Directory Structure for Linux**

If you specify `/opt/pgi` as the base directory for installation, the following directory structure is created by the PGI installation script:

<table>
<thead>
<tr>
<th>This directory...</th>
<th>Contains...</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/opt/pgi/linux86/9.0/bin</code></td>
<td>linux86 32-bit compilers &amp; tools</td>
</tr>
<tr>
<td><code>/opt/pgi/linux86/9.0/lib</code></td>
<td>linux86 32-bit runtime libraries</td>
</tr>
<tr>
<td><code>/opt/pgi/linux86/9.0/liblf</code></td>
<td>linux86 32-bit large-file support libs (used by <code>-Mlfs</code>)</td>
</tr>
<tr>
<td><code>/opt/pgi/linux86/9.0/include</code></td>
<td>linux86 32-bit header files</td>
</tr>
<tr>
<td><code>/opt/pgi/linux86-64/9.0/bin</code></td>
<td>linux86-64 compilers &amp; tools</td>
</tr>
<tr>
<td><code>/opt/pgi/linux86-64/9.0/lib</code></td>
<td>linux86-64 <code>-mcmode=small</code> libs</td>
</tr>
<tr>
<td><code>/opt/pgi/linux86-64/9.0/libso</code></td>
<td>linux86-64 <code>-fpic</code> shared libraries for <code>-mcmode=medium</code> development</td>
</tr>
</tbody>
</table>
End-user Environment Settings on Linux

This directory... | Contains...
--- | ---
/opt/pgi/linux86-64/9.0/include | linux86-64 header files
/opt/pgi/linux86/9.0/REDIST
/opt/pgi/linux86-64/9.0/REDIST | Re-distributable runtime libraries
/opt/pgi/linux86/9.0/EXAMPLES
/opt/pgi/linux86-64/9.0/EXAMPLES | Compiler examples
/opt/pgi/linux86/9.0/doc
/opt/pgi/linux86-64/9.0/doc | Documentation
/opt/pgi/linux86/9.0/man
/opt/pgi/linux86-64/9.0/man | UNIX-style man pages
/opt/pgi/linux86/9.0/jre
/opt/pgi/linux86-64/9.0/jre | JAVA environment for PGDBG and PGPROF graphical user interfaces
/opt/pgi/linux86/9.0/src
/opt/pgi/linux86-64/9.0/src | Fortran 90 source files for included modules
/opt/pgi/linux86/9.0/mpi/mpich
/opt/pgi/linux86-64/9.0/mpi/mpich | MPICH1 scripts and libraries.

Additionally, a network install creates the following directories:

### Table 4.2. Linux Directory Structure Sample

This directory... | Contains...
--- | ---
/opt/pgi/linux86/9.0/lib-linux86-g | linux86 32-bit libpgc library dependent on the version of glibc installed on each machine
/opt/pgi/linux86/9.0/include-g | linux86 32-bit header files dependent on the version of glibc or gcc installed on each machine
/opt/pgi/linux86-64/9.0/include-g | linux86-64 64-bit header files dependent on the version of glibc or gcc installed on each machine

**End-user Environment Settings on Linux**

Now that you have installed the compilers in, for example, /opt/pgi, an end-user of the PGI compilers and tools must initialize their shell environment to use the compilers successfully. Assume the license file is in /opt/pgi/license.dat, and the lmgrd license manager is running.

**Note**

Each user must issue the following sequence of commands to initialize the shell environment before using the PGI compilers and tools.

To make the 64-bit PGI compilers and tools the default:

In csh, use these commands:
% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/linux86-64/9.0/man
% setenv LM_LICENSE_FILE $PGI/license.dat
% set path = ($PGI/linux86-64/9.0/bin $path)

Once the 64-bit compilers are the defaults, you can make the MPICH commands and MPICH man pages accessible using these csh commands:

% set path = ($PGI/linux86-64/9.0/mpi/mpich/bin $path)
% setenv MANPATH "$MANPATH":$PGI/linux86-64/9.0/mpi/mpich/man

In bash, sh, or ksh, use these commands:

$ PGI=/opt/pgi; export PGI
$ MANPATH=$MANPATH:$PGI/linux86-64/9.0/man; export MANPATH
$ LM_LICENSE_FILE=$PGI/license.dat; export LM_LICENSE_FILE
$ PATH=$PGI/linux86-64/9.0/bin:$PATH; export PATH

Once the 64-bit compilers are the defaults, you can make the MPICH commands and MPICH man pages accessible using these commands.

$ PATH=$PGI/linux86-64/9.0/mpi/mpich/bin:$PATH
$ export PATH
$ MANPATH=$MANPATH:$PGI/linux86-64/9.0/mpi/mpich/man
$ export MANPATH

To make the 32-bit PGI compilers and tools the default:

In csh, use these commands:

% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/linux86/9.0/man
% setenv LM_LICENSE_FILE $PGI/license.dat
% set path = ($PGI/linux86/9.0/bin $path)

Once the 32-bit compilers are the defaults, you can make MPICH commands and MPICH man pages accessible using these csh commands.

% set path = ($PGI/linux86/9.0/mpi/mpich/bin $path)
% setenv MANPATH "$MANPATH":$PGI/linux86/9.0/mpi/mpich/man

In bash, sh, or ksh, use these commands:

$ PGI=/opt/pgi; export PGI
$ MANPATH=$MANPATH:$PGI/linux86/9.0/man; export MANPATH
$ LM_LICENSE_FILE=$PGI/license.dat; export LM_LICENSE_FILE
$ PATH=$PGI/linux86/9.0/bin:$PATH; export PATH

Once the 32-bit compilers are the defaults, you can make MPICH commands and MPICH man pages accessible using these commands.

$ PATH=$PGI/linux86/9.0/mpi/mpich/bin:$PATH; export PATH
$ MANPATH= $MANPATH:$PGI/linux86/9.0/mpi/mpich/man; export MANPATH

Common Linux Installation Issues

If you are having problems with the installation, check out these tips.
Java Runtime Environment (JRE)

Although the PGI installation on Linux includes a 32-bit version of the Java Runtime Environment (JRE), sufficient 32-bit X Windows support must be available on the system for the JRE and the PGI software that depends on it to function properly. On some systems, notably recent releases of Fedora Core, these libraries are not part of the standard installation. The required X Windows support generally includes these libraries:

libXau   libXdmcp   libxcb   libX11   libXext
Chapter 5. Installations on Windows, SFU, and SUA

This section describes how to install PGI Workstation on a Windows system. These instructions are valid for Win64, Win32, SFU, SUA32, and SUA64 platforms, though supported platforms vary by product. Further, this installation information is applicable to both permanent and trial installations.

For installations on 64-bit x64 systems running a 64-bit operating system, the PGI installer installs the 64-bit and 32-bit versions of the PGI compilers and tools. For installations on 32-bit x86 systems, the PGI installer installs only the 32-bit versions of the PGI compilers and tools.

Preparing to Install on Windows, SFU, SUA

PGI Workstation for Windows and SFU/SUA includes the Microsoft Open Tools, essential tools and libraries required to compile, link, and execute programs on Windows. No additional Microsoft tools or libraries are needed. The Microsoft Open Tools includes a subset of the full Microsoft Platform SDK. PGI Workstation 9.0 can also compile and link against the Microsoft Platform SDK. For information about how to download the Platform SDK, visit http://msdn.microsoft.com/platformsdk.

For SFU and SUA installations, the GNU SDK and GNU utilities must be installed.

For SUA, you must download the Utilities and Software Development Kit (SDK) for UNIX-based Applications from www.microsoft.com and install the Base SDK and GNU SDK.

For SFU, at installation time you must select both the "Interix SDK" and "Interix GNU SDK" components from the Custom Installation.

Note

For more information on Windows interoperability with Unix and Linux Systems, go to www.interopsystems.com.
Installing Required Utilities for SUA

For SUA installations, the GNU SDK and GNU utilities must be installed. You must download the Utilities and Software Development Kit (SDK) for UNIX-based Applications from www.microsoft.com and install the Base SDK and GNU SDK.

To install the Base SDK and GNU SDK, follow these steps.

1. Verify SUA is installed on your system.
3. Type SDK for SUA in the Search box and click Go.
4. Click Download details: Utilities and Software Development Kit (SDK) for UNIX-based Applications.
5. Read the description and scroll down to the download area, selecting the file that is most appropriate for your system.
6. Download the appropriate package.

   Tip

Save the file in your desktop or to My Documents so you can recover the disk space once the software is loaded.

7. Navigate to where you stored the downloaded package, double-click the installer to unzip the package to a temporary directory, and then click Next.
8. Accept the license agreement, and click Next.
9. Select Custom Installation, and click Next.

   Note

   The standard installation does not provide a complete UNIX development environment. In the Custom Installation allows you to obtain all the GNU compilers, build tools, and other SDK components.

10. Select all available components, and click Next.
11. Read the information in the GNU SDK dialog, which describes the GNU Library General Public License (LGPL), and click Next.
12. In the Security Settings dialog, check these two boxes, and click Next.

   Enable setup behavior for SUA programs
   Change the default behavior to case sensitive.

13. Click Finish to complete the installation.

Installing Required Utilities for SFU

For SFU installations, the GNU SDK and GNU utilities must be installed. In addition, at installation time you must select the Custom Installation and install both the "Interix SDK" and "Interix GNU SDK" components.
To install SFU:

2. Type Window Services for Unix in the Search box and click Go.
3. Click Download details: Windows Services for UNIX Version 3.5.
4. Read the description and scroll down to the download area, clicking Download to start the download.

   **Tip**

   Save the file in your desktop or to My Documents so you can recover the disk space once the software is loaded.

5. In the location where you saved the file, double-click it to unzip it, again, unzipping it into your My Documents area so you can recover the space.
6. Run My Documents\SFU\SfuSetup.msi.
7. When you see the Welcome dialog box, click Next.
8. Fill in or use the default for the Customer Information, and click Next.
9. Accept the license agreement and click Next.
10. Select Custom Installation and click Next.
11. Select the software components to install:

    Utilities -> Base Utilities
    Utilities -- UNIX Perl
    Interix GNU Components -- Interix GNU Utilities
    Interix GNU Components -- Interix GNU SDK
    No NFS
    No Password Synchronization
    Remote Connectivity -- Windows Remote Shell Service
    No Authentication tools for NFS
    Interix SDK
    ActiveState Perl

12. When you see the Interix GNU SDK dialog, read the information about the GNU Library General Public License (LGPL), and click Next.
13. Accept the ActiveState Perl License and Support agreement.
14. In the User Name Mapping dialog, check these two boxes, and click Next.

    Local User Name Mapping Server
    Network Information Service [NIS].

15. Configure the Local User Name Mapping using NIS by providing a Windows domain name, such as PGI.
16. In the Installation location, select the default location, which is SFU directory on your system drive, typically: C:\SFU.
17. Click Next to begin the installation.

18. When the Windows Services for Unix installation is complete, click Finish.

Tip

You may wish to remove the installation folder created when you unpacked the software package so that you can recover that disk space.

Figure 5.1. PGI Installation on Windows Overview
Once all the required utilities are installed, in preparation for installing PGI Workstation for Windows, review Figure 5.1 that shows an overview of the Windows installation process.

Note

You must install the software prior to getting the licenses.

Installation Steps for Windows, SFU, SUA

Once you have prepared for the installation, follow these instructions to install the software:

1. Have the software available and log on as Administrator.

   Note

   Administrator privileges are required to install PGI Workstation.

   Download the software from http://www.pgroup.com or another electronic distribution site.

2. Start the installation.

   If you obtained PGI Workstation from PGI electronically, run the installation executable file on the target machine. The installation executables are:

<table>
<thead>
<tr>
<th>Executable</th>
<th>Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>pgiws-901.exe</td>
<td>32-bit Windows</td>
</tr>
<tr>
<td>pgiwsx64-901.exe</td>
<td>64-bit/32-bit Windows</td>
</tr>
<tr>
<td>pgiws-sfu-901.exe</td>
<td>32-bit SFU</td>
</tr>
<tr>
<td>pgiws-sua-901.exe</td>
<td>32-bit SUA</td>
</tr>
<tr>
<td>pgiwsx64-sua-901.exe</td>
<td>64-bit/32-bit SUA</td>
</tr>
</tbody>
</table>

Setup Tool For Win32 or Win64:

The Win32/Win64 installer runs the PGI Licensing Setup tool at the conclusion of the install process. This tool automates steps three (3) through six (6) in this process. To use the PGI Licensing Setup tool, your computer must meet these requirements:

- Be connected to the Internet.
- Have Microsoft .NET Framework 2.0 (or higher) installed.
- Have Internet Explorer Version 6.0 or higher installed and configured.

If your computer does not meet these prerequisites, then you can either make any necessary changes to your system and run the PGI Licensing Setup tool later, or follow steps three through six in this process. If you choose to run the PGI Licensing Setup tool later, you can find it in the PGI Workstation folder under the Start menu.

When you run the program, the PGI Licensing Setup tool walks you through the license installation process. It prompts you for your PGI web account (Email address) and password, as well as the type of license you wish to install, such as a 2-week trial license or a permanent license. The tool also informs you of any problems it experiences with installation and provides instructions on how to proceed.
Note

If the PGI Licensing Setup tool reports a successful license installation, then skip to step 7.


When installation executable has completed, execute the following commands to make the PGI products accessible and to initialize your environment for use of FLEXnet.

**For Win32 or Win64:**

Whenever you bring up a PGI command window by double-left-clicking on the PGI Workstation desktop icon, PGI compilers and tools are accessible and your environment pre-initialized. Proceed to Step 4.

**For SFU, SUA32 or SUA64:**

Bring up an SFU, SUA32 or SUA64 shell window. In the commands that follow, substitute `<platform>` with sfu32, sua32 or sua64 respectively, depending on the platform on which you are installing:

In csh, use these commands:

```plaintext
% setenv PGI /opt/pgi
% set path=/opt/pgi/<platform>/9.0/bin $path
% setenv MANPATH "$MANPATH":/opt/pgi/<platform>/9.0/man
% setenv LM_LICENSE_FILE "$LM_LICENSE_FILE":/opt/pgi/license.dat
```

In bash, sh, or ksh, use these commands:

```plaintext
$ PATH=/opt/pgi/<platform>/9.0/bin:$PATH; export PATH
$ MANPATH=$MANPATH:/opt/pgi/<platform>/9.0/man; export MANPATH
$ LM_LICENSE_FILE= $LM_LICENSE_FILE:/opt/pgi/license.dat
$ export LM_LICENSE_FILE
```

You should add these commands to your shell startup files to ensure that you have access to the PGI products upon future shell invocations.


Verify the release number of the products you have installed. On Win32 or Win64 systems, open PGI Workstation from your desktop by double-left-clicking the PGI icon and read the first line displayed in the BASH shell window.

To verify the release number of PGI products you have installed on SFU, SUA32 or SUA64, open a shell window with your environment and path initialized as outlined in Step 3, and then issue the following commands:

- For Fortran 77, use: `pgf77 -V x.f`
- For Fortran 95, use: `pgf95 -V x.f`
- For HPF, use: `pghpf -V x.f`
- For C++, use: `pgCC -V x.c`
- For ANSI C, use: `pgcc -V x.c`
Note

These commands can be successfully executed even if the files \texttt{x.f} or \texttt{x.c} do not exist and you have not completed the licensing phase of the installation. The PGI compiler drivers support this as a quick check that you have installed the proper version of the compilers and have initialized your environment to enable access to that version.

5. Generate and Install license keys.

All of the PGI compilers and tools are license-managed and require installation of license keys to make the PGI software operational.

To obtain license keys, you need the following information:

- An account on the PGI website.

  You probably created this account when you downloaded the PGI software.

- If you purchased a license without creating an account, one was created for you when your order was processed. Please check for an activation email from webmaster@pgroup.com

- If you don’t have an account, you can create one at:
  
  https://www.pgroup.com/account/register.php

- The FLEXlm hostid and hostname of the computer on which the software is installed, which is echoed to your screen by the installer.

Note

After you have installed the PGI software, you can obtain your FLEXlm hostid by double-left clicking on the PGI Workstation desktop icon to open a PGI Workstation command window, or opening an SFU, SUA32 or SUA64 shell window, and typing these commands:

```
PGI$ cd $PGI
PGI$ cat license.info
```

You should see information similar to the following:

For a permanent license, please read the order acknowledgment that you received. Connect to www.pgroup.com/login with the registered email and password in your order acknowledgment.

```
FLEXlm Host ID: 001234A98765
Installation: C:\Program Files\PGI\  
PGI Release: 9.0
```

You can also obtain evaluation license keys for the PGI products by connecting to the URL www.pgroup.com/login with a web browser and logging in using an existing PGI web-user account or creating a new account if you have not previously registered with PGI.
Generate License Keys

Once you have logged on at www.pgroup.com/login using your PGI web-user email and password, you will see a screen similar to the following:

Welcome

Use the links below to manage your PGI web-user account.

Download software - For updating or evaluation.
Create permanent keys - For purchased PGI products.
Create trial keys - For a two-week evaluation of the PGI product of your choice.
Display a PIN code - Use your old (pre-2009) PIN-based username and password to display the new PIN code.
Tie a PIN to this account - Use a PIN code to tie a PIN to your account and create permanent license keys.
Manage password - Change your webuser account password or request a temporary password if you've forgotten it.
Update profile - Modify your account information.
FAQ - Answers to common questions.

To generate permanent license keys:

1. Click Create permanent keys.

2. Click the PIN associated with the product for which you wish to generate license keys. If you don’t see any PINs listed, you first need to tie one to your account.
   a. Obtain your PIN code from your original PGI order confirmation email.
   b. Click the link: Tie a PIN to this account.
   c. Follow the instructions provided.

3. Click License keys to generate the keys.

To generate trial license keys:

1. Click Create trial keys.

2. Accept the terms of the agreement.

3. Enter the hostid exactly as it appears in the message that is displayed during installation or when you issue the command:

   PGI$ cd $PGI
   PGI$ cat license.info

4. Click the Generate license key button.

Install License Keys

Once you have generated your trial or permanent license keys, copy and paste them into the file /opt/pgi/license.dat, or substitute the appropriate installation directory path if you have not installed in the default /opt/pgi directory.
For example, if you have purchased PGI Fortran Workstation for Linux, the license.dat file should look similar to the following:

Install License Keys

Once you have generated your trial or permanent license, cut and paste them into your license.dat file. In a typical configuration, where C:\ is the system drive and you installed the software using the default location, this file would be found in:

C:\Program Files\PGI\license.dat

- If you have not received license keys from PGI previously, replace the contents of the license.dat file created during installation with the PGI Workstation keys you generated above.
- If your license.dat file already contains keys you have received from PGI previously, overwrite the keys if they are for a previous release of PGI Workstation or PGI Server; append the new keys to the keys already in this file if they are for a different PGI product, such as PGI Visual Fortran.
- If you are evaluating PGI Workstation with trial keys, skip to Step 5. You do not need to start the license server.

6. Start the PGI License Server.

The FLEXnet license system requires that a license server be running. The installation process creates a Windows Service called PGI License Server. As soon as a valid license.dat file is in place, as described in Step 5, this service can be started.

**Important**

You do not need to start the license server with trial keys.

The PGI License Server is a Windows Service. Therefore, to start it, do this:

1. Open the Services dialog from the Start menu:

   Start | Control Panel | Administrative Tools | Services

2. Select PGI License Server.

3. Select Start.

**Note**

The PGI License Server service starts automatically on system reboot, provided that the license.dat file contains valid keys.

7. Review Documentation.

You can view the online HTML and PDF documentation using any web browser by opening the file: www.pgroup.com/resources/docs.htm
Customizing the Command Window

By default, when you double-left-click on the PGI Workstation desktop icon, a standard black-background command window appears on your screen. This window is pre-initialized with environment and path settings for use of the PGI Workstation compilers and tools. If you prefer different background or text colors, font style, window size, or scrolling capability, you can customize the “shortcut” that creates the PGI Workstation command window.

To customize your window, right-click the PGI Workstation desktop icon, and select “Properties” from the pop-up menu. In the PGI Workstation Properties dialog box, select the tabs for the features you want to customize, and make the desired modifications.

PGI Workstation Directory Structure

This section contains information about the directory structure that the installation process builds.

Default Installation Directories

The PGI Workstation default installation directory depends on your platform. The following table lists the default installation directory by platform.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Default Installation Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Win32</td>
<td>%SYSTEMDRIVE%\Program Files\PGI\win32\9.0-4\</td>
</tr>
<tr>
<td>Win64</td>
<td>%SYSTEMDRIVE%\Program Files\PGI\win64\9.0-4\</td>
</tr>
<tr>
<td></td>
<td>%SYSTEMDRIVE%\Program Files (x86)\PGI\win32\9.0-4\</td>
</tr>
<tr>
<td>SFU</td>
<td>/opt/pgi/sfu32/9.0-4/</td>
</tr>
<tr>
<td>SUA64</td>
<td>/opt/pgi/sua64/9.0-4/</td>
</tr>
<tr>
<td>SUA32</td>
<td>/opt/pgi/sua32/9.0-4/</td>
</tr>
</tbody>
</table>

In addition to the product directories, the Microsoft Open Tools and, optionally, cygwin, are installed here for Win32 and Win64 installations.

%SYSTEMDRIVE%\Program Files\PGI\Microsoft Open Tools 9
%SYSTEMDRIVE%\cygwin

For SFU, SUA32, and SUA64 installations, the Microsoft Open Tools are installed here:
Note

The cygwin directory is not installed with PGI Workstation for SFU and SUA. Instead, you can use any SUA or SFU shell.

PGI Workstation Directories on a Win64 System

The following directory structure is created during installation on a Win64 system; C is your system drive.

Table 5.2. Win64 Sample Directory Structure

<table>
<thead>
<tr>
<th>Name of directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\Program Files\PGI\win64\9.0\bin</td>
<td>PGI Workstation 9.0</td>
</tr>
<tr>
<td>C:\Program Files (x86)\PGI\win32\9.0\bin</td>
<td>compilers and tools binaries</td>
</tr>
<tr>
<td>C:\Program Files\PGI\win64\9.0\lib</td>
<td>PGI Workstation 9.0 runtime and support libraries</td>
</tr>
<tr>
<td>C:\Program Files (x86)\PGI\win32\9.0\lib</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\PGI\win64\9.0\include</td>
<td>PGI Workstation 9.0 header files</td>
</tr>
<tr>
<td>C:\Program Files (x86)\PGI\win32\9.0\include</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\PGI\win64\9.0\REDIST</td>
<td>Re-distributable runtime libraries</td>
</tr>
<tr>
<td>C:\Program Files (x86)\PGI\win32\9.0\REDIST</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\PGI\win64\9.0\doc</td>
<td>Documentation</td>
</tr>
<tr>
<td>C:\Program Files (x86)\PGI\win32\9.0\doc</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\PGI\win64\9.0\man</td>
<td>Man pages for commands</td>
</tr>
<tr>
<td>C:\Program Files (x86)\PGI\win32\9.0\man</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\PGI\Microsoft Open Tools 9</td>
<td>Microsoft tools</td>
</tr>
<tr>
<td>C:\cygwin</td>
<td>Cygwin tools</td>
</tr>
</tbody>
</table>

PGI Workstation Directories on a Win32 System

The following directory structure is created during installation on a Win32 system. C is your system drive.

Table 5.3. Win32 Sample Directory Structure

<table>
<thead>
<tr>
<th>Name of directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\Program Files\PGI\win32\9.0\bin</td>
<td>PGI Workstation 9.0 compilers and tools binaries</td>
</tr>
<tr>
<td>C:\Program Files\PGI\win32\9.0\lib</td>
<td>PGI Workstation 9.0 runtime and support libraries</td>
</tr>
<tr>
<td>C:\Program Files\PGI\win32\9.0\include</td>
<td>PGI Workstation 9.0 header files</td>
</tr>
<tr>
<td>C:\Program Files\PGI\win32\9.0\REDIST</td>
<td>Re-distributable runtime libraries</td>
</tr>
<tr>
<td>C:\Program Files\PGI\win32\9.0\doc</td>
<td>Documentation</td>
</tr>
<tr>
<td>C:\Program Files\PGI\win32\9.0\man</td>
<td>Man pages for commands</td>
</tr>
<tr>
<td>C:\cygwin</td>
<td>Cygwin tools</td>
</tr>
</tbody>
</table>
### PGI Workstation Directories on an SFU System

The following directory structure is created during installation on an SFU system:

<table>
<thead>
<tr>
<th>Name of directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>/opt/pgi/sfu32/9.0/bin</td>
<td>sfu32 32-bit compilers &amp; tools</td>
</tr>
<tr>
<td>/opt/pgi/sfu32/9.0/lib</td>
<td>sfu32 32-bit runtime libraries</td>
</tr>
<tr>
<td>/opt/pgi/sfu32/9.0/include</td>
<td>sfu32 32-bit header files</td>
</tr>
<tr>
<td>/opt/pgi/sfu32/9.0/doc</td>
<td>Documentation</td>
</tr>
<tr>
<td>/opt/pgi/sfu32/9.0/man</td>
<td>UNIX-style man pages</td>
</tr>
<tr>
<td>/opt/pgi/sfu32/9.0/src</td>
<td>Fortran 90 source files for included modules.</td>
</tr>
</tbody>
</table>

### PGI Workstation Directories on an SUA32 System

The following directory structure is created during installation on an SUA 32-bit system:

<table>
<thead>
<tr>
<th>Name of directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>/opt/pgi/sua32/9.0/bin</td>
<td>sua 32-bit compilers &amp; tools</td>
</tr>
<tr>
<td>/opt/pgi/sua32/9.0/lib</td>
<td>sua 32-bit runtime libraries</td>
</tr>
<tr>
<td>/opt/pgi/sua32/9.0/include</td>
<td>sua 32-bit header files</td>
</tr>
<tr>
<td>/opt/pgi/sua32/9.0/doc</td>
<td>Documentation</td>
</tr>
<tr>
<td>/opt/pgi/sua32/9.0/man</td>
<td>UNIX-style man pages</td>
</tr>
<tr>
<td>/opt/pgi/sua32/9.0/src</td>
<td>Fortran 90 source files for included modules.</td>
</tr>
</tbody>
</table>

### PGI Workstation Directories on an SUA64 System

The following directory structure is created during installation on an SUA 64-bit system:

<table>
<thead>
<tr>
<th>Name of directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>/opt/pgi/sua64/9.0/bin</td>
<td>sua 64-bit compilers &amp; tools</td>
</tr>
</tbody>
</table>
### End-user Environment Settings for SFU and SUA

Now that you have installed the compilers in, for example, /opt/pgi, an end-user of the PGI compilers and tools must initialize their shell environment to use the compilers successfully. Assume the license file is in place and the PGI License Service is running.

**Note**

Each user must issue the following sequence of commands to initialize the shell environment before using the PGI compilers and tools.

In the commands below, substitute `<platform>` with `sfu32`, `sua32` or `sua64` respectively, depending on the platform on which you are using:

**In csh, use these commands:**

```
% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/<platform>/9.0/man
% set path=(PSGI/<platform>/9.0/bin $path)
```

**In bash, sh, or ksh, use these commands:**

```
$ PGI=/opt/pgi; export PGI
$ MANPATH=$MANPATH:$PGI/<platform>/9.0/man; export MANPATH
$ PATH=$PGI/<platform>/9.0/bin:$PATH; export PATH
```

### LM_LICENSE_FILE and FLEXLM_BATCH

This section describes two environment variables that you can use with FLEXnet:

- **LM_LICENSE_FILE**
- **FLEXLM_BATCH**

#### LM_LICENSE_FILE

The system environment variable `LM_LICENSE_FILE` is not required by PGI Workstation on Windows but you can use it to override the default location that is searched for the license.dat file.

To use the system environment variable `LM_LICENSE_FILE`, set it to the full path of the license key file. To do this, follow these steps:
1. Open the System Properties dialog (Start | Control Panel | System).

2. Select the ‘Advanced’ tab.

3. Click the ‘Environment Variables’ button.

   • If LM_LICENSE_FILE is not already an environment variable, create a new system variable for it. Set its value to the full path, including the name of the file, for the license key file.

   • If LM_LICENSE_FILE already exists as an environment variable, append the path to the license file to the variable’s current value using a semi-colon to separate entries.

FLEXLM_BATCH

By default, on Windows, the license server creates interactive pop-up messages to issue warning and errors. This behavior is controlled by the environment variable FLEXLM_BATCH. Although it is not recommended, you can prevent interactive pop-ups from appearing. To do this, set the environment variable FLEXLM_BATCH to 1.

Windows Firewall and PGI Debugging

Microsoft Windows Firewall runs by default on some types of systems, such as Windows XP and Windows Vista. If this firewall is running on your system, it may try to block the PGI debug engine when you debug a program. If this happens, Windows Firewall displays a dialog that allows you to unblock the application named pgdebug. If you choose the Unblock option, Windows Firewall adds pgdebug to its list of Exceptions and you do not see the pop-up dialog again unless you reinstall.

On rare occasions, this pop-up may prevent your very first debug session from starting correctly. If this occurs, use the Task Manager to end the debug session.

To prevent any Firewall issues, you can manually add pgdebug to the Windows Firewall list of Exceptions. To do this, follow these instructions:

1. From the Control Panel, open your system’s Security Center. If your system does not have a Security Center, then you do not have Windows Firewall.

2. In the Windows Security Center, examine the Windows Firewall settings.

   If the Firewall is OFF, no further action is required.

   If you do not see the setting, proceed to the next step.

3. In the Manage Security Settings for: section, click Windows Firewall.

4. Select the Exceptions tab in the Windows Firewall dialog box.

The next steps depend on your OS.

   • If your OS is 32-bit, the final step is to add the 32-bit pgdebug to the Exceptions list.

   • If your OS is 64-bit, the final step is to add both the 32-bit pgdebug and the 64-bit pgdebug to the Exceptions list.
Note

The remaining instructions assume the system drive is C.

32-bit OS

Add the 32-bit **pgdebug** to the Exceptions list.

1. Click Add Program…
2. Use the Browse button to navigate to
   
   C:\Program Files\PGI\win32\9.0-2\bin\pgdebug.exe.
3. Select pgdebug.exe.
4. Click Open.
5. Click OK.

64-bit OS

Add both the 32-bit **pgdebug** and the 64-bit **pgdebug** to the Exceptions list.

1. Add the 32-bit pgdebug to the Exceptions list:
   a. Click Add Program…
   b. Use the Browse button to navigate to
      
      C:\Program Files (x86)\PGI\win32\9.0-4\bin.
   c. Select pgdebug.exe.
   d. Click Open.
   e. Click OK.

2. Add the 64-bit pgdebug to the Exceptions list.
   a. Click Add Program…
   b. Use the Browse button to navigate to
      
      C:\Program Files\PGI\win64\9.0-2\bin.
   c. Select pgdebug.exe.
   d. Click Open.
   e. Click OK.

Note

Once you have completed the preceding process for your OS, the Firewall no longer blocks PGI debugging.
Common Windows Installation Problems

The most common installation problems on Windows are related to licensing.

To troubleshoot your installation, first check that the license.dat file you are using contains valid license keys. Second, check that the PGI License Server, a Windows Service, has been started.

Typical FLEXnet errors encountered may include the following:

- When using a PGI compiler or tool, a FLEXible License Manager dialog appears that states ‘LICENSE MANAGER PROBLEM: No such feature exists.’

  This message may appear because the license.dat file accessed by the FLEXnet License Manager does not contain valid license keys.

- When using a PGI compiler or tool, a FLEXible License Manager dialog appears that states ‘LICENSE MANAGER PROBLEM: Cannot connect to license server system.’

  This message may appear because the PGI License Server has not been started.

- When starting the PGI License Server, a system message appears that states ‘The PGI License Server service on Local Computer started and then stopped. Some services stop automatically if they have no work to do, for example, the Performance Logs and Alerts service.’

  This message may appear because the license.dat file accessed by the FLEXnet License Manager does not contain valid license keys.

- You may see the following message: ‘LICENSE MANAGER PROBLEM: Failed to checkout license’

  This message may appear because the PGI License Server has not been started.

- By default, on Windows, the license server creates interactive pop-up messages to issue warning and errors. You can use the environment variable FLEXLM_BATCH to prevent interactive pop-up windows.

  To do this, set the environment variable FLEXLM_BATCH to 1.

- On SFU, if the compilers get segmentation faults or produce core dumps, the problem might be related to Windows Data Execution Prevention.

  The solution to these errors is to modify the system boot.ini file to set /noexecute=AlwaysOff; and then reboot.

  For more information, refer to this online resource for SFU: www.interopsystems.com.

For assistance with difficulties related to the installation, send email to trs@pgroup.com.
Chapter 6. Installations on APPLE Mac OS X

This section describes how to install PGI Workstation on an Apple system. It covers local installs, and is applicable to permanent or trial installations.

Note

PGI Workstation and PGI Server for Mac OS are only supported on Intel Core and Core 2 Duo processors running Mac OS X version 10.4.11 or greater. Previous versions of Mac OS may be unstable for 64-bit programs. Using this release requires that Apple Xcode 2.5 or greater be installed. Xcode is available from http://developer.apple.com.

• For installations on 32-bit x86 systems, the PGI installation process installs only the osx86 versions of the PGI compilers and tools.
• For installations on 64-bit x64 systems running an osx86-64 execution and development environment, the PGI installation process installs the osx86-64 version of the PGI compilers and tools.
• If the 32-bit gcc development package is already installed on the system, the 32-bit osx86 tools are installed on a 64-bit x64 system.

The 32-bit and 64-bit compilers, tools, and supporting components have the same command names, and the environment you target by default, either osx86-64 or osx86, depends on the version of the compiler that comes first in your path settings.

The default installation base directory is /opt/PGI.

Preparing to Install on Apple Mac OS X

To prepare for the installation:

• Verify you have enough free disk space.
  • On the osx86 platform, PGI installation requires 250 MB of free disk space.
  • On the osx86-64 platform, PGI installation requires 500 MB of freedisk space.
Installation Steps for Mac OS

- Verify that Xcode 2.5 or greater is installed.
  - If you know how to run Xcode, start Xcode and click About Xcode to verify the version is 2.5 or greater.
  - If you do not know how to run Xcode or are uncertain if it is installed on your system, do the following:
    1. From the Apple Menu, select About This Mac.
    2. Click More Info.
    4. Scroll through the alphabetical list and verify Xcode is in it.
    5. Verify the version is 2.5 or greater.

Note

PGI Workstation for Mac OS requires the Xcode application, which provides several components of the tool chain, including the system assembler, linker, and runtime libraries. However, PGI Workstation runs in a Terminal, not in the Xcode IDE, and the PGDBG debugger and PGPROF profiler open Java windows.

In preparation for installing PGI Workstation on Mac OS, first review Figure 6.1, an overview of the installation process.

Installation Steps for Mac OS

Once you have prepared for the installation, follow these instructions to install the software:

1. Access the installation package. Download the software from http://www.pgroup.com or another electronic distribution site. The file you download is pgiosx-901.dmg. If it didn’t mount automatically, open this file to mount it.

2. Install the software.

   Click PGI Workstation 9.0-2.pkg, which is part of the mounted disk. Follow the installation instructions.

   1. When you see the initial system check dialog, click continue to allow the install script to check that your system has the required components for installing the software, such as Xcode 2.5 or greater and gcc.

   2. Select the hard drive and, if you want, you can also select an installation directory other than the default directory: /opt/pgi.

After the software is installed, the install script performs some system-specific customization and then initializes for licensing.
3. Make PGI products accessible and prepare for licensing.

When the license file is in place, execute the following commands to make the products you purchased accessible, and to initialize your environment for use of FLEXnet. These commands assume that you use the default installation directory: /opt/pgi

Note

Substitute osx86 for osx86-64 if you are installing on a 32-bit Mac OS system or wish to target osx86 as the default.
In bash, sh, or ksh, use these commands:

```
$ PGI=/opt/pgi; export PGI
$ PATH=/opt/pgi/osx86-64/9.0/bin:$PATH; export PATH
$ MANPATH=$MANPATH:/opt/pgi/osx86-64/9.0/man; export MANPATH
$ LM_LICENSE_FILE=$LM_LICENSE_FILE:/opt/pgi/license.dat
$ export LM_LICENSE_FILE
```

In csh, use these commands:

```
% setenv PGI /opt/pgi
% set path=(/opt/pgi/osx86-64/9.0/bin $path)
% setenv MANPATH "$MANPATH":/opt/pgi/osx86-64/9.0/man
% setenv LM_LICENSE_FILE "$LM_LICENSE_FILE":/opt/pgi/license.dat
```

You should add these commands to your Terminal startup files, such as .bash_profile, to ensure that you have access to the PGI products upon future logins.

4. Verify the release number.

To verify the release number of the products you have installed, use the `-V` option on any of the compiler commands, as illustrated in the following examples. If you use `–v` instead, you also see the sequence of steps the compiler uses to compile and link programs for execution on your system.

For Fortran 77, use: `pgf77 -V x.f`
For Fortran 95, use: `pgf95 -V x.f`
For C++, use: `pgCC -V x.c`
For ANSI C, use: `pgcc -V x.c`

**Note**

These commands can be successfully executed even if the files x.f or x.c do not exist and you have not completed the licensing phase of the installation. The PGI compiler drivers support this as a quick check that you have installed the proper version of the compilers and have initialized your environment to enable access to that version.

5. Generate and Install License Keys.

All of the PGI compilers and tools are license-managed and require installation of license keys to make the PGI software operational. To obtain license keys, you need the following information:

- An account on the PGI website. You probably created this account when you downloaded the PGI software.
- If you purchased a license without creating an account, one was created for you when your order was processed. Please check for an activation email from webmaster@pgroup.com.
- If you don’t have an account, you can create one at: https://www.pgroup.com/account/register.php
- The FLEXlm hostid and hostname of the computer on which the software is installed, which is echoed to your screen by the installer.
Note

You can also obtain your FLEXnet hostid by using the following command after you have installed the products and initialized the environment variables:

```
% lmutil lmhostid
```

You see a message similar to the following one, in which 12345678abcd is unique to your machine.

The FLEXnet host ID of this machine is "12345678abcd".

To locate your hostname, use this command:

```
% lmutil lmhostid - hostname
```

Generate License Keys

Once you have logged on at www.pgroup.com/login using your PGI web-user email and password, you will see a screen similar to the following:

Welcome

Use the links below to manage your PGI web-user account.

- Download software - For updating or evaluation.
- Create permanent keys - For purchased PGI products.
- Create trial keys - For a two-week evaluation of the PGI product of your choice.
- Display a PIN code - Use your old (pre-2008) PIN-based username and password to display the new PIN code.
- Tie a PIN to this account - Use a PIN code to tie a PIN to your account and create permanent license keys.
- Manage password - Change your webuser account password or request a temporary password if you’ve forgotten it.
- Update profile - Modify your account information.
- FAQ - Answers to common questions.

To generate permanent license keys:

1. Click Create permanent keys.

2. Click the PIN associated with the product for which you wish to generate license keys. If you don’t see any PINs listed, you first need to tie one to your account.
   a. Obtain your PIN code from your original PGI order confirmation email.
   b. Click the link: Tie a PIN to this account.
   c. Follow the instructions provided.

3. Click License keys to generate the keys.

To generate trial license keys:

1. Click Create trial keys.

2. Accept the terms of the agreement.
3. Enter the hostid exactly as it appears in the message that is displayed during installation or when you issue the command:

```
% lmutil lmhostid
```

4. Click the **Generate license key** button.

## Install License Keys

Once you have generated your trial or permanent license keys, copy and paste them into the file `/opt/pgi/license.dat`, or substitute the appropriate installation directory path if you have not installed in the default `/opt/pgi` directory.

For example, if you have purchased PGI Workstation for MacOS, the `license.dat` file should look similar to the following:

```
SERVER <hostname> <hostid> 27000
DAEMON pgroupd
PACKAGE pg72-workstation-complete \
  pgroupd 9.0 2A64FB18DEBE COMPONENTS= \
"pgi-f95-osx64:9.0 pgi-f77-osx64:9.0 \
pigi-hpf-osx64:9.0 pgi-cc-osx64:9.0 \
pigi-cpp-osx64:9.0 \
pigi-f95-osx32:9.0 pgi-f77-osx32:9.0 \
pigi-hpf-osx32:9.0 pgi-cc-osx32:9.0 \
pigi-cpp-osx32:9.0 \
pigi-prof:9.0 pgi-dbgs:9.0 pgi-dbgs-gui:9.0" \
OPTIONS=SUITE_RESERVED SIGN= \
"0D91 C4A7 3569 5A7C 65EE 91EC A9AA 706C \
5EB8 C207 2DE8 9086 3926 4F84 BAF4 1C9D \
EC65 00BF 51A2 9A8D DF53 C460 7C10 663F \
9C41 92B9 F9B2 6150 D6C1 A942"
FEATURE pgi-workstation-complete pgroupd 9.0 \
  31-dec-0 5 8B15A3507BE4 \
  VENDOR_STRING=107209:8:ws DUP_GROUP=U \
  SUITE_DUP_GROUP=U BORROW=336 TS_OK SIGN= \
"0501 9540 017D 9D8C CE97 E26B F967 A848 \
864E 6ECD 2705 E535 3E6B B6B6 7ED3 0F83 \
8FDC 48FE 3AD5 E600 5B2E 26E5 D18C 2E6B \
BC30 1838 1526 C3BF F081 D19F"
```

In your license file:

- `<hostid>` should match the *hostid* you submitted above when you generated your license keys.
- If necessary, you *can* enter or edit the `<hostname>` entry manually, but you *cannot* edit the `<hostid>` entry or you will invalidate the license keys.
- The six digits immediately following the `=` in the feature line component, `VENDOR_STRING=107209:8` represent the Product ID Number (PIN) for this installation, in this example 107209. You have a similar unique PIN for your installation.

6. Start the license manager daemon.
Important

If you are evaluating PGI Workstation with trial keys, you do not need to perform this step and can proceed to Step 7.

Follow these steps to start the license server and pgroupd license daemon running on your system:

1. Use the `sudo` command to get a root bash shell; enter your password:

   ```bash
   % sudo bash
   <password>
   ```

2. Create the directory `/Library/StartupItems/PGI`:

   ```bash
   % mkdir /Library/StartupItems/PGI
   ```

3. Copy the PGI license configuration files:

   ```bash
   % cp /opt/pgi/PGI /Library/StartupItems/PGI
   % cp /opt/pgi/StartupParameters.plist /Library/StartupItems/PGI/StartupParameters.plist
   ```

4. Start the license server:

   ```bash
   % cd /Library/StartupItems/PGI
   % ./PGI start
   ```

   The license server should now be running. It should restart automatically whenever you reboot.

Installation of your FLEXnet-style licensing of our products is now complete. For assistance with difficulties related to the installation, send email to trs@pgroup.com.

7. Review Documentation.

You can view the online HTML and PDF documentation using any web browser by opening the file:

- `opt/pgi/osx86/9.0/doc/index.htm`
- `/opt/pgi/osx86-64/9.0/doc/index.htm`

You may want to bookmark this location for easy future reference to the online manuals.

**Typical Directory Structure for Mac OS X**

If you specify `/opt/pgi` as the base directory for installation, the following directory structure is created during the PGI installation process:

<table>
<thead>
<tr>
<th>Name of directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/opt/pgi/osx86/9.0/bin</code></td>
<td>osx86 32-bit compilers &amp; tools</td>
</tr>
<tr>
<td><code>/opt/pgi/osx86/9.0/lib</code></td>
<td>osx86 32-bit runtime libraries</td>
</tr>
<tr>
<td><code>/opt/pgi/osx86/9.0/include</code></td>
<td>osx86 32-bit header files</td>
</tr>
</tbody>
</table>
End-user Environment Settings on Mac OS X

Now that you have installed the compilers in, for example, /opt/pgi, you must initialize your environment to use the compilers successfully. Each user must issue the following sequence of commands to initialize the shell environment before using the PGI compilers and tools.

Issue these commands that assume this:

- The license file is in /opt/pgi/license.dat.
- The lmgrd license manager is running.

The following commands make the 32-bit compilers the default.

In bash, sh, or ksh, use these commands:

```
$ PGI=/opt/pgi; export PGI
$ MANPATH=$MANPATH:$PGI/osx86/9.0/man; export MANPATH
$ LM_LICENSE_FILE=$PGI/license.dat; export LM_LICENSE_FILE
$ PATH=$PGI/osx86/9.0/bin:$PATH; export PATH
```

In csh, use these commands:

```
% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/osx86/9.0/man
% setenv LM_LICENSE_FILE $PGI/license.dat
% set path = ($PGI/osx86/9.0/bin $path)
```

The following commands make the 64-bit PGI compilers and tools the default.

In bash, sh, or ksh, use these commands:

```
$ PGI=/opt/pgi; export PGI
$ MANPATH=$MANPATH:$PGI/osx86-64/9.0/man; export MANPATH
$ LM_LICENSE_FILE=$PGI/license.dat; export LM_LICENSE_FILE
$ PATH=$PGI/osx86-64/9.0/bin:$PATH; export PATH
```
In csh, use these commands:

```
% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/osx86-64/9.0/man
% setenv LM_LICENSE_FILE $PGI/license.dat
% set path = ($PGI/osx86-64/9.0/bin $path)
```

To use the local MPI that comes with PGI Workstation, use these commands:

In bash, sh, or ksh, use this commands:

```
$ PATH=$PGI/osx86-64/9.0/mpi/openmpi/bin:$PATH
$ export PATH
```

In csh, use this commands:

```
% set path = ($PGI/osx86-64/9.0/mpi/openmpi/bin $path)
```
Chapter 7. Accelerator Installation and Licensing

Once PGI Release 9.0-2 or higher is installed, there are a few additional steps to use the PGI Accelerator compilers.

Licensing

If you purchased the PGI Accelerator compilers feature, then your license includes support for this feature. No separate license is required.

If you did not purchase this feature and would like to use it, please contact sales@pgroup.com

Required Files

After installing the PGI Release 9.0-2 or higher compilers and tools, you must create a sitenvrc file to tell the compilers the location of the installation directory that contains your NVIDIA CUDA software.

The sitenvrc file should contain three lines similar to the following sample. In this example the CUDA software was installed in the /opt/cuda directory.

```
set NVOPEN64DIR=/opt/cuda/open64/lib;
set CUDADIR=/opt/cuda/bin;
set CUDALIB=/opt/cuda/lib;
```

The default NVIDIA Compute Capability for generated code is 1.3. You can change the default to 1.0 by adding the following line to the sitenvrc file. Adding this line disables generation of NVIDIA GPU code for double precision operations.

```
set COMPUTECAP=10;
```

You must place the sitenvrc file in the following directory, where $PGI is the PGI installation directory, which is typically /opt/pgi or /usr/pgi.

$PGI/linux86-64/9.0-2/bin/
After creating the `sitenvrcc` file and acquiring the PGI Accelerator compilers license key, you can use the option `-ta=nvidia` with the `pgf95` or `pgcc` commands. For more information on the `-ta` flag and the suboptions that relate to the target accelerators, refer to the PGI User's Guide.

The compiler automatically invokes the necessary CUDA software tools to create the kernel code and embeds the kernels in the Linux object file.

**Note**

To access the accelerator libraries, you must link an accelerator program with the `-ta` flag as well.
Chapter 8. Use OpenMPI with PGI Profiling

PGI provides performance profiling of MPI message passing support for OpenMPI applications on Linux and Mac OS X. On Apple systems, no special configuration is necessary. On Linux systems you must configure the OpenMPI installation to work with the PGI profiling system.

Instructions for Linux

This section contains instructions on how to install and configure your system for OpenMPI profiling on Linux. The basic steps include:

1. **Build OpenMPI with PGI compilers and install.** Build the OpenMPI software distribution with PGI compilers and install it.

2. **Configure OpenMPI for PGI profiling.** For more information, refer to “Configure OpenMPI for PGI Profiling,” on page 56.

3. **Build your program.** Build using the OpenMPI compiler wrappers (mpicc, mpic++, mpif77, and/or mpif90) with one of the PGI profiling options.

   **Note**

   When you build with -Mprof=time|lines|func|hwcts, MPI profiling is automatically included.

4. **Run your program.** Run as you normally would. One or more files named pgprof.out is created in your working directory.

5. **Run the profiler.** Invoke the profiler to see the results of your profiling run.

   `pgprof -exe your_program`

Compiler Wrapper data Files

To configure your OpenMPI installation for PGI profiling, you must make a few simple modifications to some configuration files, which we refer to as compiler wrapper data files.
The compiler wrapper data files are located in the /share/openmpi directory of your OpenMPI installation. Example compiler wrapper data files located in your PGI '/etc' directory are available for you to direct modifications of the wrapper data files generated when you built OpenMPI.

The wrapper file names are:
- mpicc-wrapper-data.txt
- mpic++-wrapper-data.txt
- mpif77-wrapper-data.txt
- mpif90-wrapper-data.txt

A sample wrapper file includes a block of data similar to the following.

Note

The lines in bold are ones you modify when you configure your OpenMPI installation for PGI profiling.

```text
compiler_args=
project=Open MPI
project_short=OMPI
version=1.2.8
language=C
compiler_env=CC
compiler_flags_env=CFLAGS
compiler=pgcc
extra_includes=
preprocessor_flags=-D_REENTRANT
compiler_flags=
linker_flags= 
libs=-lmpi -lopen-rte -lopen-pal
     -lrt -ldl
     -Wl,--export-dynamic -lnsl -lutil -lpthread -ldl
required_file=
includedir=${includedir}
libdir=${libdir}
```

Configure OpenMPI for PGI Profiling

To configure OpenMPI for PGI profiling, you edit the compiler wrapper data files. The lines that you modify are in bold in the sample wrapper data file in the previous section.

Important

Before you begin, make backup copies of your original wrapper data files.

Make these modifications:

1. Add the line `compiler_args=` before any other configuration lines.
2. Copy the entire data block in the sample file twice.

You need a data block for each of these compiler options:

- `-Mprof=func|lines`
- `-Mprof=time|hwcts` (hwcts is linux86-64 only.)
3. In the second data block, modify the `compiler_args=` and the `compiler_flags=` lines. The PGI profiling options are shown just to the right of the equal sign. The compiler flags you select immediately follow the equal sign, with a space between each flag.

Your lines should look similar to these:

```
compiler_args=-Mprof=func;-Mprof=lines
... 
compiler_flags= 
... 
```

4. In the third data block, modify the `compiler_args=` line and the `compiler_flags=` lines. The PGI profiling options are shown just to the right of the equal sign. The compiler flags in this data block should include: -W0, -profile, lines at the beginning of the list of flags you select.

Your lines should look similar to these:

```
compiler_args=-Mprof=time;-Mprof=hwcts
... 
compiler_flags=-W0,-profile,lines
... 
```

5. In both the second and third data blocks, modify the `libs=` line so that `-lpgnod_prof_openmpi` comes just before `-lmpi`.

Note

Do not modify any other `lib` values.

The new `libs=` line looks similar to this:

```
libs=-lpgnod_prof_openmpi -lmpi -lopen-rte -lopen-pal -lrt -ldl
-Wl,--export-dynamic -lnsl -lutil -lpthread -ldl 
```

**Modified Compiler Wrapper Data File Sample**

When you complete your modifications, your new wrapper data file has three data blocks that look similar to these. The lines you modified are in bold.

```
compiler_args=
  project=Open MPI
  project_short=OMPI
  version=1.2.8
  language=C
  compiler_env=CC
  compiler_flags_env=CFLAGS
  compiler=pgcc
  extra_includes=
  preprocessor_flags=-D_REENTRANT
  compiler_flags=
  linker_flags=
  libs=-lmpi -lopen-rte -lopen-pal -lrt -ldl -Wl,--export-dynamic -lnsl
  -lutil -lpthread -ldl
  required_file=
  includedir=${includedir}
  libdir=${libdir}
```
compiler_args=-Mprof=func;-Mprof=lines
project=Open MPI
project_short=OMPI
version=1.2.8
language=C
compiler_env=CC
compiler_flags_env=CFLAGS
compiler=pgcc
extra_includes=
preprocessor_flags=-D_REENTRANT
compiler_flags=
linker_flags=
libs=-lpgnod_prof_openmpi
-lmpi -lopen-rte -lopen-pal -lrt -ldl
-Wl,--export-dynamic -lnsl -lutil -lpthread -ldl
required_file=
includedir=${includedir}
libdir=${libdir}

compiler_args=-Mprof=time;-Mprof=hwcts
project=Open MPI
project_short=OMPI
version=1.2.8
language=C
compiler_env=CC
compiler_flags_env=CFLAGS
compiler=pgcc
extra_includes=
preprocessor_flags=-D_REENTRANT
compiler_flags=-W0,-profile,lines
linker_flags=
libs=-lpgnod_prof_openmpi
-lmpi -lopen-rte -lopen-pal -lrt -ldl
-Wl,--export-dynamic -lnsl -lutil -lpthread -ldl
required_file=
includedir=${includedir}
libdir=${libdir}
Chapter 9. Contact Information

You can contact The Portland Group at:

The Portland Group
STMicroelectronics, Inc.
Two Centerpointe Drive
Lake Oswego, OR 97035 USA

Or contact us electronically using any of the following means:

Fax: +1-503-682-2637
Sales: sales@pgroup.com
Support: trs@pgroup.com
WWW: www.pgroup.com

The PGI User Forum is monitored by members of the PGI engineering and support teams as well as other PGI customers. The forum newsgroups may contain answers to commonly asked questions. Log in to the PGI website to access the forum:

www.pgroup.com/userforum/index.php

All technical support is by email or submissions using an online form at www.pgroup.com/support. Phone support is not currently available.

Many questions and problems can be resolved at our frequently asked questions (FAQ) site at www.pgroup.com/support/faq.htm.

PGI documentation is available at www.pgroup.com/resources/docs.htm or in your local copy of the documentation in the release directory doc/index.htm.