



PGI[®] 2012
Installation Guide

Version 12.1

The Portland Group[®]

While every precaution has been taken in the preparation of this document, The Portland Group® (PGI®), a wholly-owned subsidiary of STMicroelectronics, Inc., makes no warranty for the use of its products and assumes no responsibility for any errors that may appear, or for damages resulting from the use of the information contained herein. The Portland Group retains the right to make changes to this information at any time, without notice. The software described in this document is distributed under license from STMicroelectronics and/or The Portland Group and may be used or copied only in accordance with the terms of the end-user license agreement ("EULA").

PGI Workstation, PGI Server, PGI Accelerator, PGF95, PGF90, PGFORTRAN, and PGI Unified Binary are trademarks; and PGI, PGHPE, PGF77, PGCC, PGC++, PGI Visual Fortran, PVE, PGI CDK, Cluster Development Kit, PGPROF, PGDBG, and The Portland Group are registered trademarks of The Portland Group Incorporated. Other brands and names are property of their respective owners.

No part of this document may be reproduced or transmitted in any form or by any means, for any purpose other than the purchaser's or the end user's personal use without the express written permission of STMicroelectronics and/or The Portland Group.

PGI® 2012 Installation Guide
Copyright © 2010-2012 STMicroelectronics, Inc.
All rights reserved.

Printed in the United States of America
First Printing: Release 2012, version 12.1, January 2012

Technical support: trs@pgroup.com
Sales: sales@pgroup.com
Web: www.pgroup.com

Contents

1. Release 2012 Introduction	1
Product Overview	1
PGI Workstation and PGI Server Comparison	1
PGI CDK Cluster Development Kit	1
Release Components	2
Terms and Definitions	3
Supported Processors	3
Supported Operating Systems	5
Product Support	7
2. PGI 2012 Installation Overview	9
Before You Begin	9
PGI Server Network Installations	10
PGI Installations on Linux - gcc Compiler Requirements	10
gcc Compiler Availability	11
PGI CDK Installation Overview	11
Cluster Configurations	12
Open Source Component Overview	13
3. Licensing	15
Licensing Terminology	15
License Options	15
Permanent and Trial License Keys	15
License Keys and System Configurations	16
PGI Workstation Licensing	16
PGI Server Licensing	16
PGI CDK Licensing	16
The FlexNet License Manager	17
License Support	17
4. PGI Installations on Linux	19
Prepare to Install PGI Workstation or PGI Server on Linux	19

Preparing to Install PGI CDK on Linux	22
Installation Steps for Linux	24
Typical Directory Structure for Linux	33
End-user Environment Settings	34
PGI Workstation and PGI Server	34
PGI CDK	35
Common Linux Installation Issues	37
Java Runtime Environment (JRE)	37
5. Installations on Apple Mac OS X	39
Preparing to Install on MacOS	39
Installation Steps for MacOS	40
Typical Directory Structure for MacOS	46
End-user Environment Settings on MacOS	46
6. Installations on Microsoft Windows	49
Preparing to Install on Windows	49
Installation Steps for Windows	49
Customizing the Command Window	54
PGI Workstation Directory Structure	54
Default Installation Directories	54
PGI Workstation Directories on a 64-bit Windows System	55
PGI Workstation Directories on a 32-bit Windows 'System	55
LM_LICENSE_FILE and FLEXLM_BATCH	56
LM_LICENSE_FILE	56
FLEXLM_BATCH	57
Windows Firewall and PGI Debugging	57
Automatic Additions to Windows Firewall Exceptions List	57
Common Windows Installation Problems	58
7. Contact Information	59

Figures

4.1. Linux Installation Overview	20
4.2. PGI CDK Linux Installation Overview	23
5.1. MacOS Installation Overview	41

Tables

1.1. Processors Supported by PGI 2012	4
1.2. Operating Systems and Features Supported in PGI 2012	5
4.1. Baseline PGI Installation Linux Directory Structure	33
4.2. PGI Server Linux Directory Structure Additions	34
5.1. MacOS Directory Structure	46
6.1. Default Windows Installation Directories by Platform	54
6.2. 64-bit Windows Directory Structure	55
6.3. 32-bit Windows Directory Structure	56

Chapter 1. Release 2012

Introduction

Welcome to Release 2012 of *PGI Workstation*[™], *PGI Server*[™], and *PGI CDK*[®] *Cluster Development Kit*[®], a set of compilers and development tools for 32-bit and 64-bit x86-compatible processor-based workstations, servers, and clusters running versions of the Linux and Microsoft Windows operating systems. *PGI Workstation* and *PGI Server* are also available on Apple Mac OS X operating systems.

This installation information applies to all PGI products from The Portland Group.

Product Overview

The *PGI Workstation*, *PGI Server*, and the *PGI CDK* include exactly the same compilers and tool software. The difference is the manner in which the license enables the software.

PGI Workstation and PGI Server Comparison

- All *PGI Workstation* products include 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. The product and *license server* are on the same local machine.
- *PGI Server* products are offered in configurations identical to *PGI Workstation*, but include 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 *license server*, that is, the system on which the *PGI Server* license keys are installed. There can be multiple installations of the *PGI Server* compilers and tools on machines connected to the license server; and the users can use the product concurrently, provided they are issued a license key by the license server.

PGI CDK Cluster Development Kit

A cluster is a collection of compatible computers connected by a network. The *PGI CDK* supports parallel computation on clusters of 32-bit and 64-bit x86-compatible AMD and Intel processor-based Linux and Windows workstations or servers interconnected by a TCP/IP-based network, such as Ethernet.

Support for cluster programming does not extend to clusters combining 64-bit processor-based systems with 32-bit processor-based systems, unless all are running 32-bit applications built for a common set of working x86 instructions.

Note

Compilers and libraries can be installed on other platforms not in the user's cluster, including another cluster, as long as all platforms use a common floating license server.

The *PGI CDK* supports 64-bit x64 (AMD64, Intel 64) processor-based systems, with large array addressing in PGF77, PGF95, PGFORTRAN, PGC++, and PGCC. These systems can utilize a 64-bit address space while retaining the ability to run legacy 32-bit x86 executables at full speed.

Release Components

Release 2012 includes the following components:

- PGFORTRAN™ native OpenMP and auto-parallelizing Fortran 2003 compiler.
- PGCC® native OpenMP and auto-parallelizing ANSI C99 and K&R C compiler.
- PGC++® native OpenMP and auto-parallelizing ANSI C++ compiler.
- *PGPROF*® MPI, OpenMP, and multi-thread graphical profiler.
- *PGDBG*® MPI, OpenMP, and multi-thread graphical debugger.
- MPICH MPI libraries, version 1.2.7, for both 32-bit and 64-bit development environments (Linux only).

Note

64-bit linux86-64 MPI messages are limited to <2GB size each.

- Precompiled OpenMPI library for both 32-bit and 64-bit MacOS development environments.
- A UNIX-like shell environment for 32-bit and 64-bit Windows platforms.
- FlexNet license utilities.
- Documentation in PDF and man page formats.

Additional components for PGI CDK

The *PGI CDK* for Linux also includes these components:

- MPICH2 MPI libraries, version 1.0.5p3, for both 32-bit and 64-bit development environments.
- MVAPICH MPI libraries, version 1.1, for both 32-bit and 64-bit development environments.
- ScaLAPACK linear algebra math library for distributed-memory systems, including BLACS version 1.1- (the Basic Linear Algebra Communication Subroutines) and ScaLAPACK version 1.7 for use with MPICH or MPICH2 and the PGI compilers on Linux systems with a kernel revision of 2.4.20 or higher. This is provided in both linux86 and linux86-64 versions for AMD64 or Intel 64 CPU-based installations.

Note

linux86-64 versions are limited.

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

MPI Support

You can use PGI products to develop, debug, and profile MPI applications. The MPI profiler and debugger included with *PGI Workstation* are limited to eight local processes. The MPI profiler and debugger included with *PGI Server* are limited to 16 local processes. The MPI profiler and debugger included with *PGI CDK* supports up to 256 remote processes.

Terms and Definitions

These release notes contain a number of terms and definitions with which you may or may not be familiar. If you encounter an unfamiliar term in these notes, 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, Intel Core 2 (Penryn), Intel Core i3, i5, i7 (Nehalem) processors.

Supported Processors

Table 1.1 lists the processors on which Release 2012 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 AMD64 and Intel 64-bit compatible CPUs.

PGI products have the capability to generate binaries optimized for specific AMD or Intel processors. The PGI 2012 compilers can produce PGI Unified Binary™ object or executable files containing code streams fully optimized and supported for both AMD and Intel x64 CPUs and NVIDIA GPUs.

Tip

To determine what CPU you have, use one of the following commands:

`pgf90 -V`

or

`pgcc -V`

To produce *PGI Unified Binary* files, use one of the following `-tp` command-line options, where `<target>` is any of the valid values in [Table 1.1](#).

`-tp x64`

or

`-tp <target1>,<target2>,<target3>...`

Table 1.1. Processors Supported by PGI 2012

Brand	CPU Micro-Architecture	Target Processor <code>-tp <target> [,target...]</code>	
		32-bit	64-bit
AMD			
	Opteron Bulldozer	bulldozer-32	bulldozer-64
	Opteron Six-core Istanbul	istanbul-32	istanbul-64
	Opteron Quad-core Shanghai	shanghai-32	shanghai-64
	Opteron Quad-core Barcelona	barcelona-32	barcelona-64
	Opteron Quad-core	k8-32	k8-64
	Opteron Rev E?F	k8-32	k8-64e
	Turion / Turion 64	k8-32	k8-64e
	Athlon64	NA	k8-64e
	Athlon	athlon	NA
	Athlon XP/MP	athlonxp	NA
Intel			
	Sandybridge	sandybridge-32	sandybridge-64
	Nehalem	nehalem-32	nehalem-64
	Penryn	penryn-32	penryn-64
	Core	core2-32	core2-64
	P4/Xeon EM64T	p7-32	p7-64
	Xeon Pentium4	p7-32	NA
	Pentium III	piii	NA
	Pentium II	p6	NA
Generic			
	Generic x86	p5 or px-32	NA

The compilers do work on x86 CPUs not listed in the table.

Supported Operating Systems

[Table 1.2](#) lists the operating systems, and their equivalents, on which PGI 2012 compilers and tools are supported. To determine if Release 2012 will install and run under a Linux equivalent version, such as Mandrake, Debian, CentOS, 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.

- Linux operating systems with support for x64 compatible processors 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 modern operating systems include support for Intel Hyper-threading (HT).

Most modern Linux distributions support the *Native Posix Threads Library (NPTL)*. 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 where it is available. In addition, the *PGDBG* debugger is capable of debugging executables built using either NPTL or earlier `pthread` implementations.

Many modern 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 1.2. Operating Systems and Features Supported in PGI 2012

Type	Distribution	64-bit	HT	NPTL	NUMA	glibc	GCC
Linux	RHEL 6.2	Yes	Yes	Yes	Yes	2.12	4.4.6
	RHEL 6.1	Yes	Yes	Yes	Yes	2.12	4.4.5
	RHEL 6.0	Yes	Yes	Yes	Yes	2.12	4.4.4
	RHEL 5.6	Yes	Yes	Yes	Yes	2.12	4.4.4
	RHEL 5.5	Yes	Yes	Yes	No	2.5	4.1.2
	RHEL 5.4	Yes	Yes	Yes	No	2.5	4.1.2
	RHEL 5.3	Yes	Yes	Yes	No	2.5	4.1.2
	RHEL 5.0	Yes	Yes	Yes	No	2.5	4.1.2
	RHEL 4.0	Yes	Yes	Yes	No	2.3.4	3.4.3

Type	Distribution	64-bit	HT	NPTL	NUMA	glibc	GCC
Linux	Fedora 14	Yes	Yes	Yes	Yes	2.12	4.4.5
	Fedora 13	Yes	Yes	Yes	Yes	2.12	4.4.4
	Fedora 12	Yes	Yes	Yes	Yes	2.11	4.4.2
	Fedora 11	Yes	Yes	Yes	Yes	2.9	4.3.3
	Fedora 10	Yes	Yes	Yes	Yes	2.9	4.3.2
	Fedora 9	Yes	Yes	Yes	Yes	2.8	4.3.0
	Fedora 8	Yes	Yes	Yes	Yes	2.7	4.1.2
	Fedora 7	Yes	Yes	Yes	Yes	2.6	4.1.2
	Fedora 6	Yes	Yes	Yes	Yes	2.5	4.1.1
	Fedora 5	Yes	Yes	Yes	Yes	2.4	4.1.0
	Fedora 4	Yes	Yes	Yes	No	2.3.5	4.0.0
	SuSE 11.3	Yes	Yes	Yes	Yes	2.11.2	4.5
	SuSE 11.2	Yes	Yes	Yes	Yes	2.10.1	4.4.1
	SuSE 11.1	Yes	Yes	Yes	Yes	2.9	4.3.3
	SuSE 11.0	Yes	Yes	Yes	Yes	2.8	4.3.1
	SuSE 10.3	Yes	Yes	Yes	Yes	2.6.1	4.2.1
	SuSE 10.2	Yes	Yes	Yes	Yes	2.5	4.1.0
	SuSE 10.1	Yes	Yes	Yes	Yes	2.4	4.1.0
	SuSE 10.0	Yes	Yes	Yes	Yes	2.3.5	4.0.2
	SuSE 9.3	Yes	Yes	Yes	Yes	2.3.4	3.3.5
	SLES 11	Yes	Yes	Yes	Yes	2.9	4.3.3
	SLES 10	Yes	Yes	Yes	Yes	2.4	4.1.0
	SLES 9	Yes	Yes	No	Yes	2.3.3	3.3.3
	RedHat 9.0	No	No	Yes	No	2.3.2	3.2.2
	Ubuntu 11.10	Yes	Yes	Yes	Yes	2.13.20	4.6.1
	Ubuntu 11.04	Yes	Yes	Yes	Yes	2.13.0	4.5.2
	Ubuntu 10.10	Yes	Yes	Yes	Yes	2.12.1	4.4.5
	Ubuntu 10.04	Yes	Yes	Yes	Yes	2.11.1	4.4.3
	Ubuntu 9.10	Yes	Yes	Yes	Yes	2.10.1	4.4.1
	Ubuntu 9.04	Yes	Yes	Yes	Yes	2.9	4.3.3
Ubuntu 8.10	Yes	Yes	Yes	Yes	2.8	4.3.2	
Ubuntu 8.04	Yes	Yes	Yes	Yes	2.7	4.2.1	

Type	Distribution	64-bit	HT	NPTL	NUMA	glibc	GCC
Microsoft	XP	No	Yes	NA	Yes	NA	NA
Windows	XP x64	Yes	Yes	NA	Yes	NA	NA
	2003	No	No	NA	Yes	NA	NA
	2003 x64	Yes	Yes	NA	Yes	NA	NA
	2008	No	Yes	NA	Yes	NA	NA
	2008 x64	Yes	Yes	NA	Yes	NA	NA
	Vista	No	Yes	NA	Yes	NA	NA
	Vista x64	Yes	Yes	NA	Yes	NA	NA
	Windows 7	Yes	Yes	NA	Yes	NA	NA
	HPC Server 2008	Yes	Yes	NA	Yes	NA	NA
Apple	Lion	Yes	No	NA	NA	NA	4.2.1
	Snow Leopard	Yes	No	NA	NA	NA	4.0.1

Note

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.

Linux

Java Runtime Environment (JRE)

The PGI debugger and profiler use Java SE 6. On 64-bit systems PGDBG and PGPROF use a 64-bit JRE; on 32-bit systems they use a 32-bit JRE. The correct JRE is important because PGDBG and PGPROF make calls to native code, which must be of compatible "bitness" to function properly.

Windows

PGI 2012 for Windows supports most of the features of the 32-bit and 64-bit versions for linux86 and linux86-64 environments.

Mac OS X

PGI 2012 for Mac OS X supports most of the features of the 32-bit and 64-bit versions for linux86 and linux86-64 environments. Except where noted in PGI documentation, the PGI compilers and tools on Mac OS X function identically to their Linux counterparts.

Product Support

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

The PGI Subscription Service provides the following benefits:

- 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 available at www.pgroup.com/support/support_request.php
 - Phone support is not currently available.
- Notification by e-mail 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 Subscription Service. Extensions are available in yearly increments.

Contact sales@pgroup.com if you would like information regarding the subscription service for the PGI products you have purchased.

Chapter 2. PGI 2012 Installation Overview

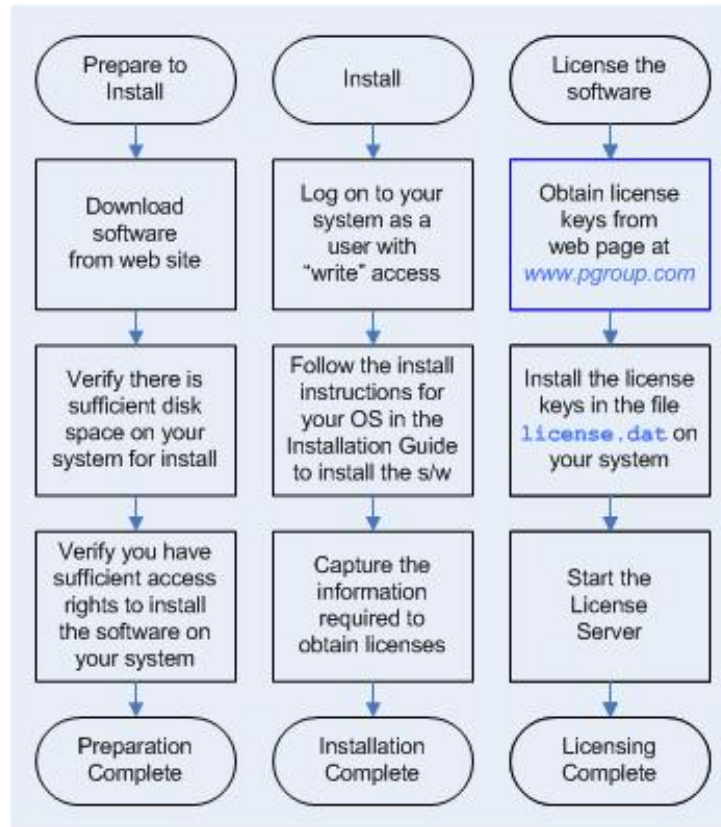
This chapter provides an overview of the steps required to successfully install *PGI Workstation*, *PGI Server*, or the *PGI CDK*. The remaining chapters provide the details of each of the steps. Specifically, “*Licensing*” describes licensing. [Chapter 4, “PGI Installations on Linux”](#) describes how to install *PGI Workstation*, *PGI Server* or *PGI CDK* in a generic manner on Linux, including how to install and run a FlexNet license daemon on Linux. [Chapter 6, “Installations on Microsoft Windows”](#) describes how to install on a Windows system, and [Chapter 5, “Installations on Apple Mac OS X”](#) describes how to install on an Apple Mac OS X system.

Before You Begin

Before you begin the installation, it is advantageous to 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
- License the Software – generating 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 PGI installation process.



For more complete information on these steps and the specific actions to take for your operating system, refer to the corresponding chapters in this guide.

PGI Server Network Installations

Linux

On Linux, *PGI Server* may be installed locally on each machine on a network or it may be installed once on a shared file system available to each machine. If you select the second method, then after the first installation you can run a simple script on each machine in the group to add each new platform to the family of machines using the common compiler installation. For more information on network installations, refer to [Chapter 4, “PGI Installations on Linux”](#).

MacOS and Windows

On MacOS and Windows, *PGI Server* must be installed locally on each machine. A network installation using a shared file system is not supported.

PGI Installations on Linux - gcc Compiler Requirements

On Linux systems, PGI compilers need the gcc tool chain (ld,as,ar) and gcc libraries and header files to be compatible with your version of Linux. Therefore, to successfully install the PGI compilers on a Linux system, the system must already have gcc compilers installed.

- To successfully install 64-bit PGI compilers, gcc 64-bit compilers must be present and in the user's path.

- To successfully install 32-bit PGI compilers, gcc 32-bit compilers must be present and in the user's path.

Note

32-bit Linux systems can only have 32-bit gcc compilers.

- To successfully install 32-bit PGI compilers on a 64-bit Linux system, gcc 32-bit compilers must be present and in the user's path.

gcc Compiler Availability

To determine if 32-bit or 64-bit gcc compilers are installed on the system, do the following:

1. Create a `hello.c` program.

```
#include <stdio.h>
int main()
{
printf(" hello\n");
}
```

2. Execute the appropriate commands to determine if you have gcc support in your path:

For 64-bit gcc support:

```
% gcc -m64 -o ./hello_64 ./hello.c ! create a 64-bit executable
% file ./hello_64 ! does the file exist, and is it 64-bit?
```

```
hello_64: ELF 64-bit LSB executable, AMD x86-64, version 1 (SYSV), for
GNU/Linux 2.6.9, dynamically linked (uses shared libs), for GNU/Linux
2.6.9, not stripped
```

For 32-bit gcc support:

```
% gcc -m32 -o ./hello_32 ./hello.c ! create a 32-bit executable
% file ./hello_32 ! does the file exist, and is it 32-bit?
```

```
hello_32: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for
GNU/Linux 2.6.9, dynamically linked (uses shared libs), for GNU/Linux
2.6.9, not stripped
```

PGI CDK Installation Overview

The following sections contain the information needed for you to successfully prepare to install the *PGI CDK* software.

As stated earlier, a cluster is a collection of compatible computers connected by a network. The *PGI CDK* software is installed on a working cluster. It is not the purpose of this product to create a cluster, or to troubleshoot one. The *PGI CDK* release can be installed on a single node, and the node can be treated as if it is a cluster.

Support for cluster programming does not extend to clusters combining AMD64 or Intel 64 CPU-based systems with IA32 CPU-based systems, unless all are running 32-bit applications built for a common set of working x86 instructions.

For multi-process programming of message-passing applications that execute on a cluster, the *PGI CDK* includes both a 32-bit and a 64-bit set of MPICH and MPICH2 libraries. These libraries include versions that can collect additional information useful in the cluster debugger *PGDBG* and the cluster profiler *PGPROF*, and they implement the MPI inter-process communication standard.

Once you have downloaded the *PGI CDK* from www.pgroup.com, you can view the documentation by loading the file `index.htm` from the directory in which you unpacked the PGI tar file. After installation, the file `index.htm` is available in the top-level PGI installation directory.

Cluster Configurations

Clusters are typically configured with two types of nodes:

- A "master" node from which jobs are launched
- "slave" nodes that are used only for computation

The *PGI CDK* is always installed by users on the master node, and the compilers and tools are installed to run from the master node only.

Typically, the master node is accessible from the general-purpose or "public" network and shares a file system with the other computers on your network using NFS. The master node and all of the slave nodes are interconnected using a second "private" network that is only accessible from computers that are part of the cluster.

There are two common cluster configurations:

1. The master node is used only for compilation and job submission, and only the slave nodes are used for computation.
2. All nodes are used for computation, including the master node.

One way to use MPICH in the first configuration is to manage job scheduling. By default, the `mpirun` command uses the master node as one of the computation nodes. It is possible to exclude the master node as a computation node in the second configuration if `mpirun` is invoked with the `-nolocal` option. For more information, refer to the man page for `mpirun`.

If you are using the first configuration, it is possible to install MPICH and run parallel MPI or HPF jobs without installing any of the other components. However, if you have multiple users running jobs on your cluster simultaneously, you should determine how to ensure your cluster nodes are allocated and used efficiently.

Typically, a master node has two network cards to allow communication to the outside network as well as to the cluster nodes themselves, which may be on their own subnet. If this is the case on your cluster, then when the installation script prompts you for the name of the master node, you should use the name associated with the network card connected to the same network as the cluster nodes.

For MPICH to run correctly, access from each node to every other node must be available via the `rsh` or `ssh` command. For example, if a 3-node cluster consists of a master, named *master*, and two slaves named *node1* and *node2*, then from *node1* as a user you should be able to issue the commands:

```
% rsh master date
% rsh node2 date
```

or

```
% ssh master date
% ssh node2 date
```

You can issue similar commands from *node2* using *master* and *node1*.

By default, all of the PGI compilers and tools will be installed on your system. You will select which of the open source components to install.

At this point, before you start the installation, you must determine:

- Which *PGI CDK* open source components – MPICH, MPICH2, and MVAPICH – you will install.
- The hostnames of all the nodes that will be included in your cluster. You need a list of these during the installation.
- The type of cluster configuration – that is, whether the master node will participate as a compute node or will be strictly a front-end for compilation, job launching, and so on.
- Whether the compute nodes can share files with the master node, which is strongly recommended.

[Chapter 4, “PGI Installations on Linux”](#) describes how to install the PGI Fortran, C and C++ compilers and tools on Linux using the *installcdk* script from PGI.

Note

Some, but not all, parts of the *PGI CDK* require root access to successfully execute the *installcdk* script.

For multi-system cluster installations, you need root access. Without root access you can create a compilers, tools, and MPICH library installation on a single local system that allows multi-process/multi-threaded development. [Chapter 4, “PGI Installations on Linux”](#) also includes the steps for configuring and starting the FlexNet license daemon that is required to make the PGI software operational. The FlexNet license daemon enables use of the PGI compilers and tools by any user on any system networked to the system on which the PGI software is installed. For example, users can compile, debug, and profile using the *PGI CDK* compilers and tools on any system on your general-purpose network, subject to the constraints on concurrent usage for the product you have purchased.

The PGI compilers and tools are license-managed, which is described in the next chapter. Further, the [“Installation Steps for Linux,” on page 24](#) provides specific information about how to use your personalized account to generate trial or permanent license keys.

Open Source Component Overview

The *PGI Compiler User’s Guide* describes basic usage of the open source components of the *PGI CDK*, including MPICH, ScaLAPACK libraries, and the example benchmark programs and tutorials.

MPICH, MPICH2, MVAPICH, and ScaLAPACK are all open source software packages that are not formally supported by The Portland Group. All source code for these components is included in the *cdk* subdirectory. Along with the source code, each of these components has end-user and implementer documentation, generally in the form of printable PostScript. Support for these products is generally provided by their respective user communities, which you can learn more about at the following URLs:

- MPICH – www.mcs.anl.gov/research/projects/mpi/mpich1 contains a wealth of information, including online documentation, tutorials, FAQ files, patch distributions, and information on how to submit bug reports to the MPICH developers.
- MPICH2 – www.mcs.anl.gov/research/projects/mpi/mpich2 contains a wealth of information, including online documentation, tutorials, FAQ files, patch distributions, and information on how to submit bug reports to the MPICH2 developers.
- MVAPICH – MVAPICH 1.1 supports many features for high performance, scalability, portability, and fault tolerance. It also supports a wide range of platforms. mvapich.cse.ohio-state.edu/overview/mvapich contains a wealth of information, including online documentation, tutorials, FAQ files, patch distributions, and how to submit bug reports to the MVAPICH developers.

Note

To use the *PGI CDK* version of MVAPICH, the prerequisite OpenFabrics (OFED) software must be installed. See www.openfabrics.org for details on OFED.

- ScaLAPACK - www.netlib.org/scalapack contains FAQ files and current distributions of ScaLAPACK

Chapter 3. Licensing

The PGI compilers and tools are license-managed. PGI software licensing uses the FlexNet Publisher (FNP) license management system from Flexera Software (www.flexerasoftware.com).

Licensing Terminology

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 software 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.

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

License Options

PGI offers licenses for either x64+GPU or x64 only platforms. *PGI Accelerator* products, the x64+GPU platform products, include support for the directive-based PGI Accelerator programming model, CUDA Fortran and PGI CUDA-x86. PGI Accelerator compilers are supported on all Intel and AMD x64 processor-based systems with CUDA-enabled NVIDIA GPUs running Linux, Mac OS X, or Windows.

Permanent and Trial License Keys

PGI 2012 includes the PGI License Setup tool to help automate your license retrieval and installation process. Use this tool to obtain your license keys, either trial or permanent.

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.

- **Permanent License Keys** – When you purchase a *permanent* PGI license, your e-mail order confirmation includes complete instructions for logging in to the pgroup.com web page and generating permanent license keys.
- **Trial License Keys** – When you register for a *trial* license, you generate trial keys using the web page: www.pgroup.com/login.php.

Note

At the conclusion of the 15-day trial period, the PGI compilers and tools and any executable files generated prior to the installation of permanent license keys will cease to function.

To retain functionality, any executables, object files, or libraries created using the PGI compilers with trial license keys must be recompiled once permanent license keys in place.

For more detailed information on how to obtain license keys, refer to Step 5 in “[Installation Steps for Linux](#),” on page 24.

License Keys and System Configurations

Executable files generated with permanent license keys in place are unconstrained, and 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 keys may become invalid. Please contact license@pgroup.com if you expect to reconfigure your system to ensure that you do not temporarily lose the use of your 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.

The license server for *PGI Server* can run on any of these supported systems: Linux, Windows, and MacOS.

PGI CDK Licensing

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

Note

We recommend the license servicer run on the cluster master node.

The FlexNet License Manager

As part of the process of installing the PGI compilers and tools, you install and configure the FlexNet license management software. The instructions in the following chapters of this guide describe how to configure license daemons for Linux, Windows, or MacOS, including installation and start-up of the license services, and proper initialization of the `LM_LICENSE_FILE` and, for Windows, `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 7.

Chapter 4. PGI Installations on Linux

This chapter describes how to install *PGI Workstation*, *PGI Server*, or *PGI CDK* in a generic manner on a Linux system. It covers local and network installs and is applicable for 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.

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

Prepare to Install PGI Workstation or PGI Server on Linux

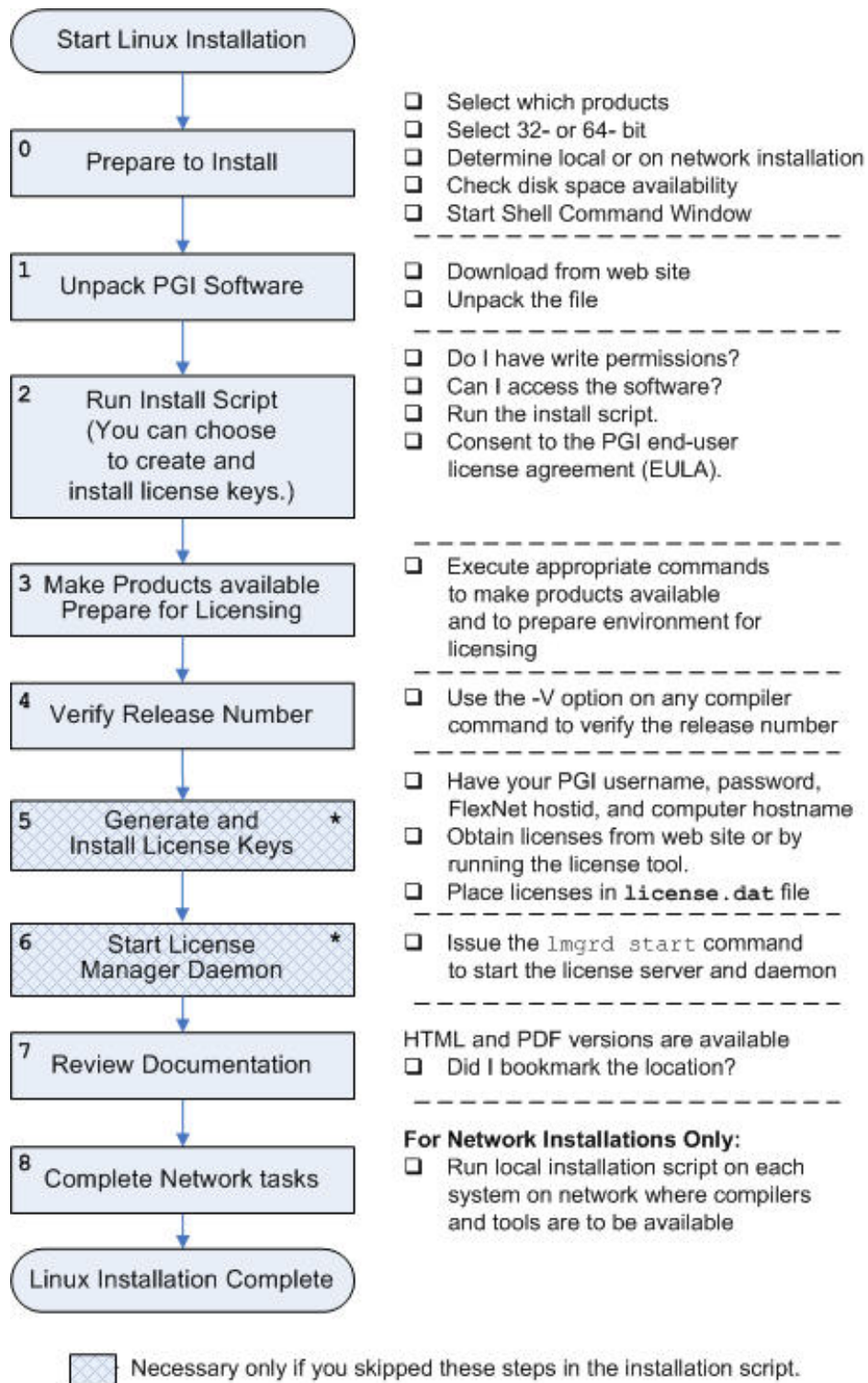
In preparation for installing *PGI Workstation* or *PGI Server* on Linux, first review the overview of the Linux installation process illustrated in [Figure 4.1](#).

Note

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

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

Figure 4.1. Linux Installation Overview



Network Installations:

If you choose to perform a network installation, 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 contains 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 running different versions of Linux.

To Prepare for the Installation:

- Locate your PGI Order Confirmation e-mail.

This e-mail contains instructions for downloading the PGI software from the PGI web site as well as other information you need for generating your permanent license. It also contains your PGI Product Identification Number (PIN) 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 for the PGI installation.
 - The uncompressed installation package requires 700 MB of free disk space.
 - The linux86 platform requires up to 450 MB of free disk space, depending on the number of packages installed.
 - The linux86-64 platform requires up to 1.4 GB of free disk space, depending on the number of packages installed.
- If you plan to use the PGI license tool, know how your computer accesses the Internet – directly or through some sort of proxy server connection.

Note

If this computer is behind a firewall at your site, make sure it can access the Internet.

If a proxy server is used, you need this additional information:

- The address (URL) of the proxy server.
- Whether the proxy server requires authentication and if so, the username and password.
- To generate license keys, log in to and account on the PGI website at `www.pgroup.com/login`. From this page you may also create an account or reset your password.
- In order to debug 32-bit applications on 64-bit Linux systems, the 32-bit `ncurses` libraries must be installed. The name of this package varies across the Linux distributions, but it contains a file with a name like `libncurses.so.5`.

Preparing to Install PGI CDK on Linux

To prepare for the installation:

1. Download the software.

Unlike other PGI software, the *PGI CDK* is not generally available. To download the software, first log in to your PGI account at www.pgroup.com/login. Once you are logged in, click the Download Software link. If you don't see the *PGI CDK* listed as a product option on the downloads page, contact PGI at sales@pgroup.com.

2. 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.

3. Log in as root on the master node to install and verify your cluster.

Note

If you fail to do this, software is installed only on the master node.

4. Verify you have enough free disk space.

- The uncompressed installation package requires 700 MB of free disk space.
- The linux86 platform requires up to 450 MB of free disk space, depending on the number of packages installed.
- The linux86-64 platform requires up to 1.4 GB of free disk space, depending on the number of packages installed.

5. Install a version of Python. (Optional)

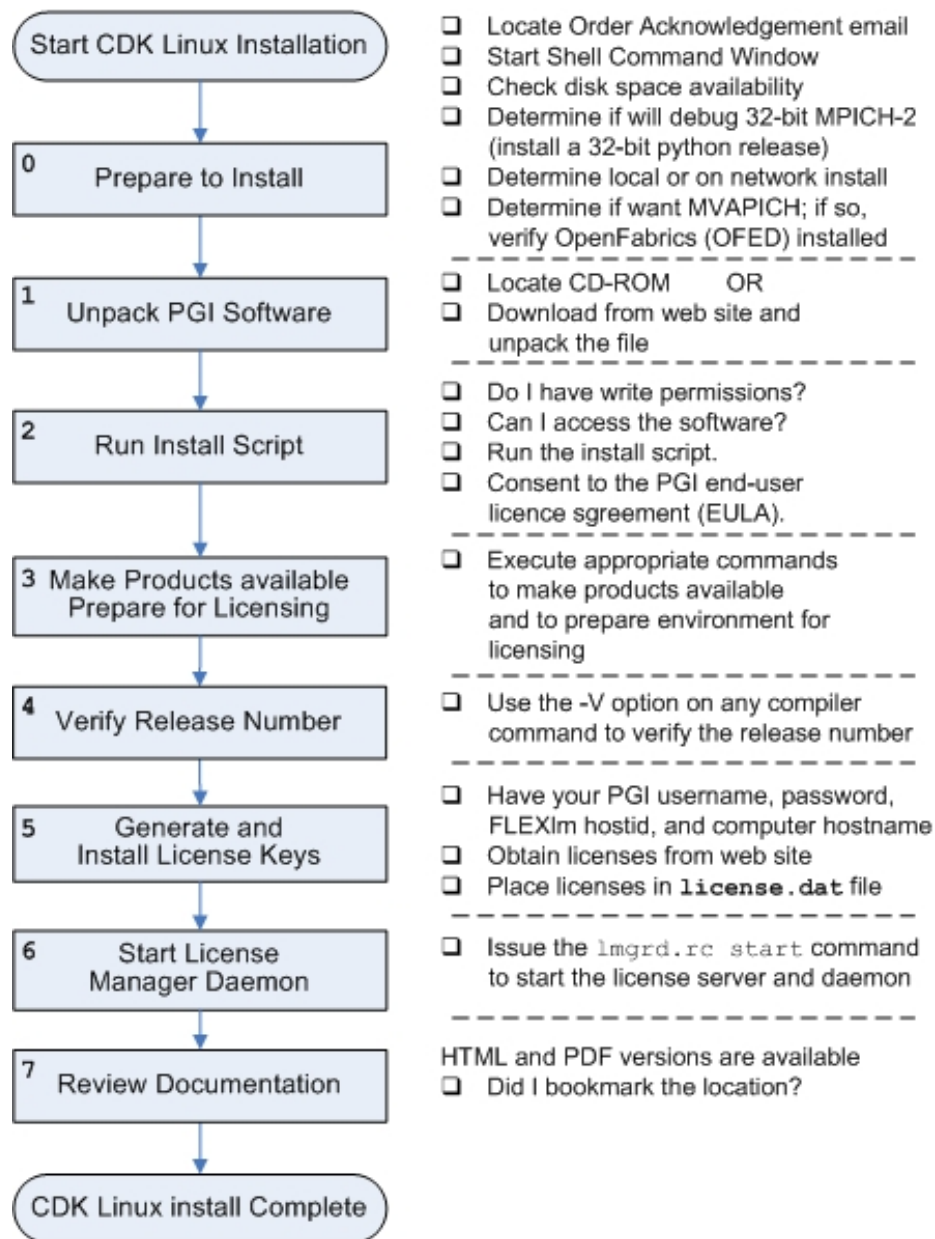
If you want to debug 32-bit MPICH-2 programs on a 64-bit system, a 32-bit Python release must be installed on the 64-bit system. The *PGI CDK* includes a pre-compiled 32-bit Python in `$PGI/linux86/12.1/mpi2/mpich/Python32` (where `$PGI` is the PGI installation directory), but it is recommended that you install a version of Python built specifically for your system.

6. Install MVAICH. (Optional)

If you want to install MVAICH, verify that you already have OpenFabrics (OFED) software installed on your system. MVAICH is an InfiniBand library. See section 2.2 for more information.

Before you begin the actual installation, review the *PGI CDK* Linux installation process illustrated in [Figure 4.2](#).

Figure 4.2. PGI CDK Linux Installation Overview



Be certain that you know this information:

- Locate your PGI order confirmation e-mail.

This e-mail 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 keys. It also contains your PGI Product Identification Number (PIN) that you need if you contact PGI.

- Know how your computer accesses the Internet – directly or through some sort of proxy server connection.

Note

If this computer is behind a firewall at your site, make sure it can access the Internet.

If a proxy is used, you need this additional information:

- The address (URL) of the proxy server.
- Whether the proxy server requires authentication and if so, the username and password.
- To log in to the PGI website to generate license keys, you need either an account (username and password) or a PIN code from the PGI order confirmation e-mail message from PGI Sales. The PIN code in the e-mail is in the section labeled: 'SHARING ADMINISTRATION.'
- To generate license keys, log in to and account on the PGI website at www.pgroup.com/login. From this page you may also create an account or reset your password.

Installation Steps for Linux

Follow these instructions to install the software:

1. Unpack the PGI software.

Download the software from www.pgroup.com or another electronic distribution site. In the instructions that follow, replace <tarfile> with the name of the file that you 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 installation script.

Note

The installation script for *PGI Workstation* and *PGI Server* is **install** and for the *PGI CDK* is **installcdk**. These instructions refer to these collectively as the installation script.

The installation script *must* run to completion to properly install the software.

Note

- If you are updating a previous release or wish to reinstall the *PGI CDK*, we recommend that you run the *uninstallcdk* script before running *installcdk*.
- If you are logged in as root, the entire cluster can be configured.
- If you are not logged in as root, the *PGI CDK* compilers and tools and MPICH, MPICH2, and MVAPICH libraries can be installed on your current machine; and be configured for that machine only. This will create a single node cluster and/or a cross-development installation.

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

```
% ./install
or
% ./installcdk
```

To successfully run this script to completion, do the following:

- Consent to the PGI end-user license agreement (EULA).
- If you are installing *PGI Server*, determine whether to perform a local installation or a network installation.
- Determine whether to install the optional components:
 - ACML math library
 - NVIDIA components
 - MPICH
- Define where to place the installation directory.
- Determine whether to use the built-in utility to generate license keys.

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

If you are performing a network installation for *PGI Server*, you are asked for a common local directory on each system. The network installation uses this local directory, or creates it if it does not exist, on each system to store system-specific header files and libraries.

The *PGI CDK* installation script installs all of the binaries for the PGI compilers and tools, MPICH, MPICH2, MVAPICH and ScaLAPACK in the \$PGI directory tree in the appropriate `bin`, `include`, `lib`, and `man` subdirectories. You are prompted for various information about how to configure your cluster as the script executes. Once the installation script has completed, exit the root shell.

Important

If you choose not to generate license keys at installation time, record the FlexNet hostid and hostname that are echoes to the screen.

Tip

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

3. Make PGI products accessible.

When the installation script has completed, execute the following commands to make the PGI products accessible and to initialize your environment for use by FlexNet.

Note

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

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/linux86-64/12.1/bin $path)
% setenv MANPATH "$MANPATH":/opt/pgi/linux86-64/12.1/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/linux86-64/12.1/bin:$PATH; export PATH
$ MANPATH=$MANPATH:/opt/pgi/linux86-64/12.1/man; export MANPATH
$ LM_LICENSE_FILE=$LM_LICENSE_FILE:/opt/pgi/license.dat;export LM_LICENSE_FILE
```

For linux86:

To use only the linux86 version of the compilers and tools, or to target 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/linux86/12.1/bin $path)
% setenv MANPATH "$MANPATH":/opt/pgi/linux86/12.1/man
% setenv LM_LICENSE_FILE "$LM_LICENSE_FILE":/opt/pgi/license.dat
```

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

```
$ PATH=/opt/pgi/linux86/12.1/bin:$PATH; export PATH
$ MANPATH=$MANPATH:/opt/pgi/linux86/12.1/man; export MANPATH
$ LM_LICENSE_FILE=$LM_LICENSE_FILE:/opt/pgi/license.dat;export LM_LICENSE_FILE
```

Note

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

4. Verify the release number of the installed software.

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 can also see the sequence of steps the compiler uses to compile and link programs for execution on your system.

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. Use them to 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.

Note

This step is necessary only if you chose not to allow the installation script to perform these tasks for you.

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, and ScaLAPACK 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 e-mail from `accounts-noreply@pgroup.com`.
 - If you don't have an account, you can create one at: `www.pgroup.com/register`.
 - If you purchased a *PGI CDK* license but do not see the software listed for download, then contact `trs@pgroup.com`.
- The FlexNet `hostid` and `hostname` of the computer on which the software is installed, which was 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 should see a message similar to the following with one or more `hostids` displayed. In this example you can use either `12345678abcd` or `edcba9876543`, but not both, as the `hostid`.

```
The FlexNet host ID of this machine is "12345678abcd edcba9876543".
```

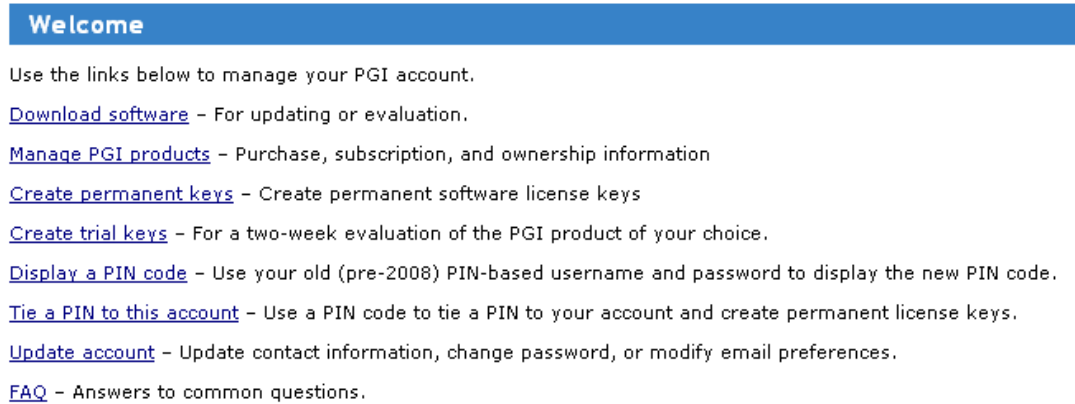
Tip

`Hostids` come from configured network cards. If you use your computer in multiple environments, you may want to run the following command in each environment to see what

hostids are configured. Then, to reduce potential license problems, choose the hostid that occurs in *all* your environments.

Generate License Keys

Log in to your PGI web account at www.pgroup.com/login. You should 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 e-mail, in the section labeled: 'SHARING ADMINISTRATION.'
 - 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
```

If multiple host ids are displayed, select any *one* of them to use as the hostid.

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.

The `license.dat` depends on the product you are installing,. However, each `license.dat` file looks somewhat similar to the following sample file from a PGI Fortran Workstation installation:

```
SERVER <hostname> <hostid> 27000
DAEMON pgroupd
PACKAGE PGI2012-workstation-complete pgroupd 2012.1231 8FE2A7F11183 \
COMPONENTS="pgi-hpf-lin64 pgi-f95-lin64 pgi-f77-lin64 \
pgi-cc-lin64 pgi-cpp-lin64 pgi-hpf-lin32 pgi-f95-lin32 \
pgi-f77-lin32 pgi-cc-lin32 pgi-cpp-lin32 pgi-hpf-lin32 \
pgi-f95-lin32 pgi-f77-lin32 pgi-cc-lin32 pgi-cpp-lin32 \
pgi-prof pgi-dbg pgi-dbg-gui " \
OPTIONS=SUITE SIGN="08AF B25A 295F BB3A CA7D \
F564 0973 BB69 2B66 187A 1BE6 6E65 BC95 2152 9E75 03BC A41D \
3422 52DF 15C4 37BB 0D7A 7F65 47E7 8B70 0F5B 987D 9A69 1FFC 2270"
FEATURE PGI2012-workstation-complete pgroupd 2012.1231 permanent 1 \
EA26405D17C1 VENDOR_STRING=123456:4:ws:accel \
HOSTID=001ec960ef1c DUP_GROUP=U SUITE_DUP_GROUP=U SIGN="0667 \
D67C 3E9E A935 287A AE93 035B 8316 CC01 F035 80CD 57E1 9F6A \
1E01 BD51 09DE 3BB6 6467 432E ADAF A9D7 E9B8 1E08 BF8F D7C5 \
E3F2 F458 8779 2420 2F97"
PACKAGE PGI71-workstation-complete pgroupd 7.1 4663E4ABD89E \
COMPONENTS="pghpf-linux86-64:7.1 pgf90-linux86-64:7.1 \
pgf77-linux86-64:7.1 pgcc-linux86-64:7.1 pgcpp-linux86-64:7.1 \
pghpf-linux86:7.1 pgf90-linux86:7.1 pgf77-linux86:7.1 \
pgcc-linux86:7.1 pgcpp-linux86:7.1 pghpf-linux86:7.1 \
pgf90-linux86:7.1 pgf77-linux86:7.1 pgcc-linux86:7.1 \
pgcpp-linux86:7.1 pgprof:7.1 pgdbg:7.1 pgdbg-gui:7.1 \
pgdbg-linux86-64:7.1" OPTIONS=SUITE SIGN="07C2 DA82 2505 0A20 \
B51E B86F D47E 2765 BB92 5EF7 0756 A652 8B80 1E2B A299 18D5 \
F82F 243B 0368 D580 9E0B 99DF A989 D202 3AAD BCC6 1A3C F517 24C6 013B"
FEATURE PGI71-workstation-complete pgroupd 7.1 permanent 1 \
35CB2A57B14C VENDOR_STRING=123456:4:ws HOSTID=001ec960ef1c \
DUP_GROUP=U SUITE_DUP_GROUP=U SIGN="023C A239 D028 AC2C 7CDF \
74C2 B32A 0E9D C68C FFAB FF8D 25CE BA74 E8C6 CCA1 076B 3560 \
4720 9A82 53F0 C1A6 1811 35D4 F8C2 A8D2 4427 4478 EB44 76E5 FCE1"
```

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 date in the file, in this example 2012.1231, represents the expiration date for your subscription service.

For example, if your subscription date for your PGI PIN (Product Identification Number) is August 1, 2012, then the date in your file is 2012.0801. For information on how to renew your subscription, refer to [“Product Support,” on page 7](#)

- The six digits immediately following the `VENDOR_STRING =` in the feature line component, 123456 of `VENDOR_STRING=123456:4` in this example, represent the PIN for this installation.

You have a similar unique PIN for your installation.

Note

Please include your PIN when contacting PGI for technical support for the products you have purchased. This PIN is also in your order confirmation e-mail.

6. Start the license manager daemon.

Important

If you used the installation script to do this or if you are evaluating PGI software with trial license 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/linux86-64/12.1/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:

```
% cd $PGI/linux86-64/12.1/bin/  
% ./lmgrd.rc start
```

Tip

To use the more restrictive access of invoking `lmgrd` with the `-local` switch, refer to the `lmgrd.rc` file for a description of the `-local` switch and how to use it.

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

```
% cd $PGI/linux86-64/12.1/bin/  
% ./lmgrd.rc stop
```

Start license server upon reboot:

To start the license server and `pgroupd` license 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/linux86/12.1/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 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.
- 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.

chkconfig(8) Utility

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

```
% cp $PGI/linux86/12.1/bin/lmgrd.rc /etc/init.d/
% /sbin/chkconfig --add lmgrd
```

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

Important

You can co-install Release 2012 with Release 2011, 2010, 9.x, 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 latest versions of `lmgrd` and `pgroupd` and a single Release 2012 license file.

If you use the `lmgrd.rc` file to start `lmgrd` automatically after a reboot of your system, you need to modify your `lmgrd` script in the `/etc/rc.d` or `/etc/init.d` directory to use the latest `lmgrd` daemon.

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

```
## Path to master daemon lmgrd
# Commented out previous path to 11.8:
#LMGRD=$PGI/<target>/11.8/bin/lmgrd
LMGRD=$PGI/<target>/12.1/bin/lmgrd

## Command to stop lmgrd
#Commented out previous path to 11.8:
#LMUTIL=$PGI/<target>/11.8/bin/lmutil
LMUTIL=$PGI/<target>/12.1/bin/lmutil
```

7. Review documentation.

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

```
$PGI/linux86-64/12.1/doc/index.htm
```

or

```
$PGI/linux86/12.1/doc/index.htm
```

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

8. Complete network installation tasks.

Note

Skip this step if you are not installing *PGI Server* 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 `/usr/pgi/shared/12.1` is the common local directory, then run the following commands on each system on the network.

```
/opt/pgi/linux86/12.1/bin/makelocalrc -x /opt/pgi/linux86/12.1 \
-net /usr/pgi/shared/12.1
/opt/pgi/linux86-64/12.1/bin/makelocalrc -x /opt/pgi/linux86-64/12.1 \
-net /usr/pgi/shared/12.1
```

These commands create a system-dependent file `localrc.machinename` in both of these directories: `/opt/pgi/linux86/12.1/bin` and `/opt/pgi/linux86-64/12.1/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/12.1/lib
/usr/pgi/shared/12.1/liblf
/usr/pgi/shared/12.1/lib64.
```

Note

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

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

The following two sections contain information detailing the directory structure of the PGI installation, and instructions for PGI end-users to 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 4.1. Baseline PGI Installation Linux Directory Structure

This directory...	Contains...
<code>/opt/pgi/linux86/12.1/bin</code>	linux86 32-bit compilers & tools
<code>/opt/pgi/linux86/12.1/lib</code>	linux86 32-bit runtime libraries
<code>/opt/pgi/linux86/12.1/liblfs</code>	linux86 32-bit large-file support libs (used by <code>-Mlfs</code>)
<code>/opt/pgi/linux86/12.1/include</code>	linux86 32-bit header files
<code>/opt/pgi/linux86-64/12.1/bin</code>	linux86-64 compilers & tools
<code>/opt/pgi/linux86-64/12.1/lib</code>	linux86-64 <code>-mcmodel=small</code> libs
<code>/opt/pgi/linux86-64/12.1/libso</code>	linux86-64 <code>-fpic</code> shared libraries for <code>-mcmodel=medium</code> development
<code>/opt/pgi/linux86-64/12.1/include</code>	linux86-64 header files
<code>/opt/pgi/linux86/12.1/REDIST</code> <code>/opt/pgi/linux86-64/12.1/REDIST</code>	Re-distributable runtime libraries
<code>/opt/pgi/linux86/12.1/EXAMPLES</code> <code>/opt/pgi/linux86-64/12.1/EXAMPLES</code>	Compiler examples
<code>/opt/pgi/linux86/12.1/doc</code> <code>/opt/pgi/linux86-64/12.1/doc</code>	Documentation
<code>/opt/pgi/linux86/12.1/man</code> <code>/opt/pgi/linux86-64/12.1/man</code>	UNIX-style man pages
<code>/opt/pgi/linux86/12.1/jre</code> <code>/opt/pgi/linux86-64/12.1/jre</code>	JAVA environment for <i>PGDBG</i> and <i>PGPROF</i> graphical user interfaces
<code>/opt/pgi/linux86/12.1/src</code> <code>/opt/pgi/linux86-64/12.1/src</code>	For Workstation and Server installations: Fortran 90 source files for included modules For PGI CDK installations: <i>PGHPF</i> MPI interface file, <code>mpi.c</code>
<code>/opt/pgi/linux86/12.1/mpi/mpich</code> <code>/opt/pgi/linux86-64/12.1/mpi/mpich</code>	MPICH1 scripts and libraries.
<code>/opt/pgi/linux86/12.1/mpi2/mpich</code> <code>/opt/pgi/linux86-64/12.1/mpi2/mpich</code>	MPICH2 scripts and libraries.
<code>/opt/pgi/linux86/12.1/mpi/mvapich</code> <code>/opt/pgi/linux86-64/12.1/mpi/mvapich</code>	MVAPICH scripts and libraries.

This directory...	Contains...
/opt/pgi/linux86/12.1/mpi/mpich/share /opt/pgi/linux86-64/12.1/mpi/mpich/share	MPICH machines .LINUX files
/opt/pgi/linux86/12.1/EXAMPLES/mpi /opt/pgi/linux86-64/12.1/EXAMPLES/mpi	MPI Examples
/opt/pgi/linux86/12.1/mpi/mpich/sbin /opt/pgi/linux86-64/12.1/mpi/mpich/sbin	MPICH <i>tstmachines</i>

Additionally, a *PGI Server* installation on a network creates the following directories:

Table 4.2. PGI Server Linux Directory Structure Additions

This directory...	Contains...
/opt/pgi/linux86/12.1/lib-linux86-g	linux86 32-bit libpgc library dependent on the version of glibc installed on each machine
/opt/pgi/linux86/12.1/include-g	linux86 32-bit header files dependent on the version of glibc or gcc installed on each machine
/opt/pgi/linux86-64/12.1/include-g	linux86-64 64-bit header files dependent on the version of glibc or gcc installed on each machine

End-user Environment Settings

PGI Workstation and PGI Server

After software installation is complete, each user's shell environment must be initialized to use the PGI compilers and tools. 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.

64-bit as default

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

In `cs`, use these commands:

```
% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/linux86-64/12.1/man
% setenv LM_LICENSE_FILE $PGI/license.dat
% set path = ($PGI/linux86-64/12.1/bin $path)
```

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


```
% set path = ($PGI/linux86-64/12.1/mpi/mpich/bin $path)
% setenv MANPATH "$MANPATH":$PGI/linux86-64/12.1/mpi/mpich/man
```

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

```
$ PGI=/opt/pgi; export PGI
$ MANPATH=$MANPATH:$PGI/linux86-64/12.1/man; export MANPATH
$ LM_LICENSE_FILE=$PGI/license.dat; export LM_LICENSE_FILE
$ PATH=$PGI/linux86-64/12.1/bin:$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=$PGI/linux86-64/12.1/mpi/mpich/bin:$PATH
$ export PATH
$ MANPATH=$MANPATH:$PGI/linux86-64/12.1/mpi/mpich/man
$ export MANPATH
```

32-bit as default

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

In csh, use these commands:

```
% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/linux86/12.1/man
% setenv LM_LICENSE_FILE $PGI/license.dat
% set path = ($PGI/linux86/12.1/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/linux86/12.1/mpi/mpich/bin $path)
% setenv MANPATH "$MANPATH":$PGI/linux86/12.1/mpi/mpich/man
```

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

```
$ PGI=/opt/pgi; export PGI
$ MANPATH=$MANPATH:$PGI/linux86/12.1/man; export MANPATH
$ LM_LICENSE_FILE=$PGI/license.dat; export LM_LICENSE_FILE
$ PATH=$PGI/linux86/12.1/bin:$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=$PGI/linux86/12.1/mpi/mpich/bin:$PATH; export PATH
$ MANPATH=$MANPATH:$PGI/linux86/12.1/mpi/mpich/man; export MANPATH
```

PGI CDK

With either the trial or permanent license keys in place, choose the applicable commands from the following lists and execute them to make the products you have purchased accessible.

Important

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

Important

For the following path settings, the installation is in the default directory: `/opt/pgi`

Set Environment Variables for Licensing

Execute the following commands to initialize your environment for use of FlexNet licensing.

In `csh`, use these commands:

```
% setenv PGI /opt/pgi
% setenv LM_LICENSE_FILE "$LM_LICENSE_FILE":/opt/pgi/license.dat
```

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

```
$ PGI=/opt/pgi; export PGI
$ LM_LICENSE_FILE=$LM_LICENSE_FILE:$PGI/license.dat; export LM_LICENSE_FILE
```

Set End-user PATH Variables

This section describes the commands required to access `MPICH`, `MPICH2`, and `MVAPICH` as well as the man pages associated with each of these.

Note

If you install only the `linux86` versions of the compilers or wish to target `linux86` as the default, use a setup similar to the `x64 linux86-64` one shown here, substituting *linux86* in place of *linux86-64* in all path settings.

MPICH access

To access `MPICH` and the `MPICH` man pages for `x64 linux86-64`, execute these commands:

In `csh`, use these commands:

```
% set path = (/opt/pgi/linux86-64/12.1/bin \
/opt/pgi/linux86-64/12.1/mpi/mpich/bin $path)
% setenv MANPATH "$MANPATH":/opt/pgi/linux86-64/12.1/mpi/mpich/man
```

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

```
$ PATH=/opt/pgi/linux86-64/12.1/bin:/opt/pgi/linux86-64/12.1/mpi/mpich/bin:$PATH
$ MANPATH=$MANPATH:/opt/pgi/linux86-64/12.1/mpi/mpich/man
$ export PATH MANPATH
```

MPICH2 access

To access `MPICH2` and the `MPICH2` man pages for `x64 linux86-64`, execute these commands:

In `csh`, use these commands:

```
% set path = (/opt/pgi/linux86-64/12.1/bin \
/opt/pgi/linux86-64/12.1/mpi2/mpich/bin $path)
% setenv MANPATH "$MANPATH":/opt/pgi/linux86-64/12.1/mpi2/mpich/man
```

In bash, sh or ksh, use these commands:

```
$ PATH=/opt/pgi/linux86-64/12.1/bin:/opt/pgi/linux86-64/12.1/mpi2/mpich/bin:$PATH
$ MANPATH=$MANPATH:/opt/pgi/linux86-64/12.1/mpi2/mpich/man
$ export PATH MANPATH
```

MVAPICH access

To access MVAPICH and the MVAPICH man pages for x64 linux86-64, execute these commands:

In csh, use these commands:

```
% set path = (/opt/pgi/linux86-64/12.1/bin \
    /opt/pgi/linux86-64/12.1/mpi/mpich/bin $path)
% setenv MANPATH "$MANPATH":/opt/pgi/linux86-64/12.1/mpi/mpich/man
```

In bash, sh or ksh, use these commands:

```
$ PATH=/opt/pgi/linux86-64/12.1/bin:/opt/pgi/linux86-64/12.1/mpi/mvapich/bin:$PATH
$ MANPATH=$MANPATH:/opt/pgi/linux86-64/12.1/mpi/mvapich/man
$ export PATH MANPATH
```

Common Linux Installation Issues

If you are having problems executing the debugger or profiler, you might want to check out the Java Runtime Environment.

Java Runtime Environment (JRE)

The PGI debugger and profiler use Java SE 6. On 64-bit systems PGDBG and PGPROF use a 64-bit JRE; on 32-bit systems they use a 32-bit JRE. The correct JRE is important because PGDBG and PGPROF make calls to native code, which must be of compatible "bitness" to function properly.

Chapter 5. Installations on Apple Mac OS X

This section describes how to install PGI software on an Apple computer running Mac OS X. It covers local installs, and is applicable to permanent or trial installations.

Note

PGI Workstation and *PGI Server* for MacOS are only supported on Intel processor-based Macintosh computers running MacOS version 10.4.11 or greater. Previous versions of MacOS 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 MacOS

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 free disk space.

- 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.
 3. Select System profiler | Software | Applications.
 4. Scroll through the alphabetical list and verify Xcode is in it.
 5. Verify the version is 2.5 or greater.

Note

PGI software for MacOS requires the Xcode application, which provides several components of the tool chain, including the system assembler, linker, and runtime libraries. However, PGI software runs in Terminal, not in the Xcode IDE, and the *PGDBG* debugger and *PGPROF* profiler have their own GUIs.

In preparation for installing PGI software on MacOS, first review [Figure 5.1](#), an overview of the installation process.

Installation Steps for MacOS

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

Note

If you intend to use CUDA Fortran or the PGI Accelerator Fortran or C compilers, then install PGI Workstation before installing PGI Accelerator.

1. Access the installation package.

Download the software from www.pgroup.com or another electronic distribution site. The file you download is `PGI-Workstation-120.dmg`. If it didn't mount automatically, open this file to mount it.

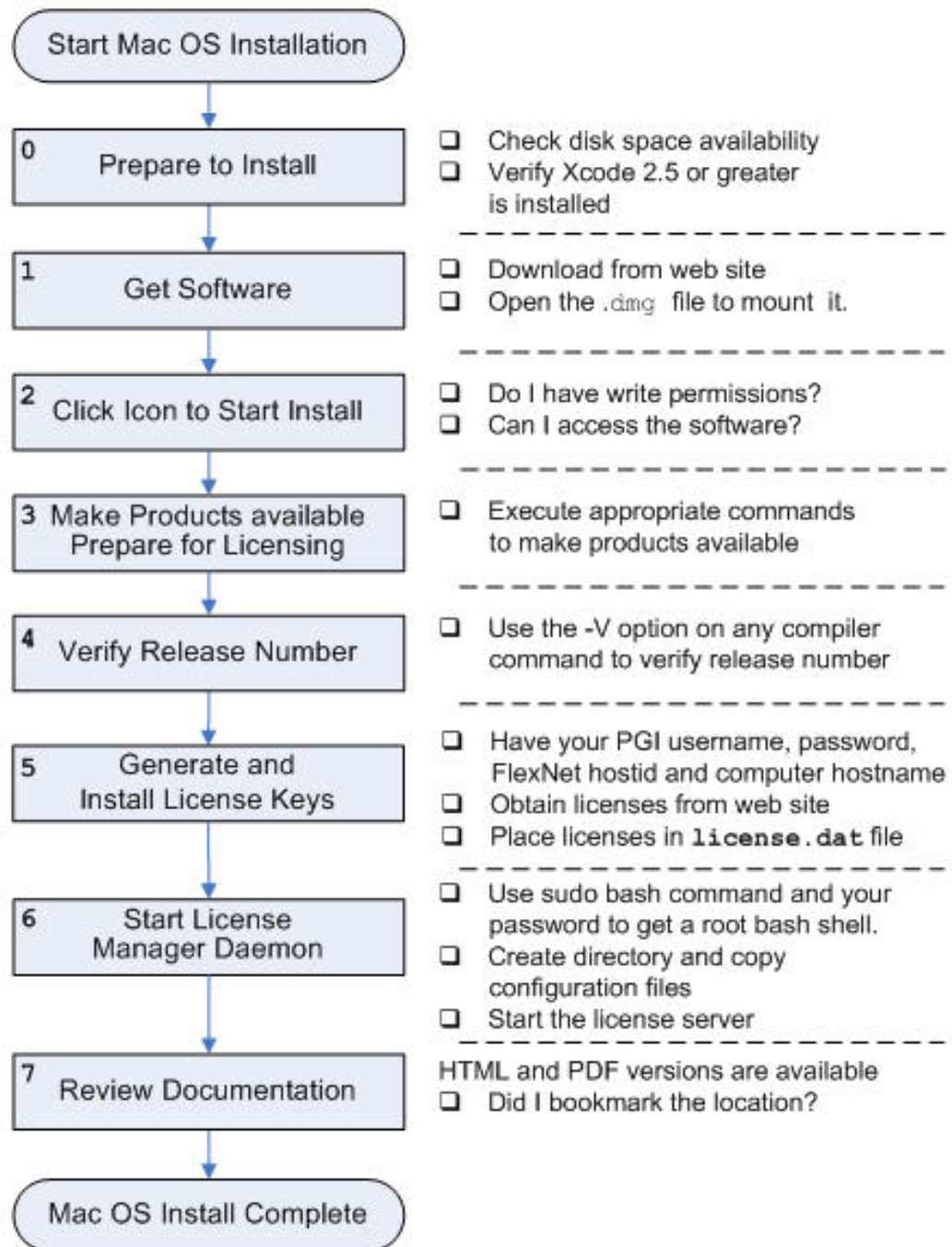
2. Install the software.

Double-click the `PGI Workstation 2012.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 installation 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 drive and, if you want, also select an installation directory other than the default directory:
`/opt/pgi`.

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

Figure 5.1. MacOS Installation Overview



3. Make PGI products accessible and prepare for licensing.

Once software installation is complete, you can bring up a PGI Terminal window by double-clicking the `PGI2012_terminal` file in the installation directory, typically `/opt/pgi`. If Terminal is currently open, you can bring up a PGI window by using the `New Windows` and `New Tab` commands, and then selecting `PGI2012`.

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 2003, use:	pgfortran -V x.f
For C++, use:	pgCC -V x.cc
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. Use it to 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 the 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 e-mail from `accounts-noreply@pgroup.com`.
- If you don't have an account, you can create one at: `www.pgroup.com/register`
- The FlexNet hostid and hostname of the computer on which the software is installed. The installer echoes this information to your screen.

Note

You can obtain your FlexNet hostid by using the following command in a PGI Terminal window.

```
% lutil lmhostid
```

You should see a message similar to the following with one or more hostids displayed.

```
The FlexNet host ID of this machine is "12345678abcd edcba9876543".
```

You can use either `12345678abcd` or `edcba9876543`, but not both, as the hostid.

Tip

Hostids come from configured network cards. If you use your computer in multiple environments, you may want to run the following command in each environment to see what hostids are configured. Then, to reduce potential license problems, choose the hostid that occurs in *all* your environments.

To locate your hostname, use this command:

```
% lmutil lmhostid -hostname
```

The above command returns output in the form: `HOSTNAME=xxxxxx`. However, when you create a license, use only the actual hostname, that is `xxxxx` in the hostname field. For example, if the information displayed is `HOSTNAME=abcd1234`, then use only `abcd1234`

Generate License Keys

Log in to your PGI web account at www.pgroup.com/login. You should see a screen similar to the following:

Welcome

Use the links below to manage your PGI account.

[Download software](#) - For updating or evaluation.

[Manage PGI products](#) - Purchase, subscription, and ownership information

[Create permanent keys](#) - Create permanent software license keys

[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.

[Update account](#) - Update contact information, change password, or modify email preferences.

[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 e-mail, in the section labeled: 'SHARING ADMINISTRATION.'
 - 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
```

If multiple host ids are displayed, select any *one* of them to use as the hostid.

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 PGI2012-workstation-complete pgroupd 2012.1231 8FE2A7F11183 \
  COMPONENTS="pgi-hpf-lin64 pgi-f95-lin64 \
  pgi-f77-lin64 pgi-cc-lin64 pgi-cpp-lin64 \
  pgi-hpf-lin32 pgi-f95-lin32 pgi-f77-lin32 \
  pgi-cc-lin32 pgi-cpp-lin32 pgi-hpf-lin32 \
  pgi-f95-lin32 pgi-f77-lin32 pgi-cc-lin3 \
  pgi-cpp-lin32 pgi-prof pgi-dbg \
  pgi-dbg-gui" OPTIONS=SUITE SIGN="08AF B25A 295F BB3A CA7D \
  F564 0973 BB69 2B66 187A 1BE6 6E65 BC95 2152 9E75 03BC A41D \
  3422 52DF 15C4 37BB 0D7A 7F65 47E7 8B70 0F5B 987D 9A69 1FFC 2270"
FEATURE PGI2012-workstation-complete pgroupd 2012.1231 permanent 1 \
  EA26405D17C1 VENDOR_STRING=123456:4:ws:accel \
  HOSTID=001ec960ef1c DUP_GROUP=U SUITE_DUP_GROUP=U SIGN="0667 \
  D67C 3E9E A935 287A AE93 035B 8316 CC01 F035 80CD 57E1 9F6A \
  1E01 BD51 09DE 3BB6 6467 432E ADAF A9D7 E9B8 1E08 BF8F D7C5 \
  E3F2 F458 8779 2420 2F97"
PACKAGE PGI71-workstation-complete pgroupd 7.1 4663E4ABD89E \
  COMPONENTS="pghpf-linux86-64:7.1 pgf90-linux86-64:7.1 \
  pgf77-linux86-64:7.1 pgcc-linux86-64:7.1 pgcpp-linux86-64:7.1 \
  pghpf-linux86:7.1 pgf90-linux86:7.1 pgf77-linux86:7.1 \
  pgcc-linux86:7.1 pgcpp-linux86:7.1 pghpf-linux86:7.1 \
  pgf90-linux86:7.1 pgf77-linux86:7.1 pgcc-linux86:7.1 \
  pgcpp-linux86:7.1 pgprof:7.1 pgdbg:7.1 pgdbg-gui:7.1 \
  pgdbg-linux86-64:7.1" OPTIONS=SUITE SIGN="07C2 DA82 2505 0A20 \
  B51E B86F D47E 2765 BB92 5EF7 0756 A652 8B80 1E2B A299 18D5 \
  F82F 243B 0368 D580 9E0B 99DF A989 D202 3AAD BCC6 1A3C F517 24C6 013B"
FEATURE PGI71-workstation-complete pgroupd 7.1 permanent 1 \
  35CB2A57B14C VENDOR_STRING=123456:4:ws HOSTID=001ec960ef1c \
  DUP_GROUP=U SUITE_DUP_GROUP=U SIGN="023C A239 D028 AC2C 7CDF \
  74C2 B32A 0E9D C68C FFAB FF8D 25CE BA74 E8C6 CCA1 076B 3560 \
  4720 9A82 53F0 C1A6 1811 35D4 F8C2 A8D2 4427 4478 EB44 76E5 FCE1"
```

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 date in the file, in this example 2012.1231, represents the expiration date for your subscription service.

For example, if your subscription date for your PGI Product Identification Number (PIN) is August 1, 2012, then the date in your file is 2012.0801. For information on how to renew your subscription, refer to [“Product Support,” on page 7](#)

- The six digits immediately following the = in the feature line component, `VENDOR_STRING=123456:4` represent the PIN for this installation, in this example 123456. 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:

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

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

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

3. Copy the PGI license configuration files:

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

4. Start the license server:

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

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

Installation of your FlexNet licensing is now complete. For assistance with difficulties related to the installation, send e-mail 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/osx86/12.1/doc/index.htm
or
/opt/pgi/osx86-64/12.1/doc/index.htm
```

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

Typical Directory Structure for MacOS

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

Table 5.1. MacOS Directory Structure

Name of directory	Contents
<code>/opt/pgi/osx86/12.1/bin</code>	osx86 32-bit compilers & tools
<code>/opt/pgi/osx86/12.1/lib</code>	osx86 32-bit runtime libraries
<code>/opt/pgi/osx86/12.1/include</code>	osx86 32-bit header files
<code>/opt/pgi/osx86/12.1/doc</code>	Documentation
<code>/opt/pgi/osx86/12.1/man</code>	UNIX-style man pages
<code>/opt/pgi/osx86/12.1/src</code>	Fortran source files for included modules.
<code>/opt/pgi/osx86/12.1/mpi</code>	osx86 32-bit OpenMPI for local development
<code>/opt/pgi/osx86-64/12.1/bin</code>	osx86-64 compilers & tools
<code>/opt/pgi/osx86-64/12.1/lib</code>	osx86-64 <code>-mmodel=small</code> libs
<code>/opt/pgi/osx86-64/12.1/include</code>	osx86-64 header files
<code>/opt/pgi/osx86-64/12.1/doc</code>	Re-distributable runtime libraries
<code>/opt/pgi/osx86-64/12.1/doc</code>	Documentation
<code>/opt/pgi/osx86-64/12.1/man</code>	UNIX-style man pages
<code>/opt/pgi/osx86-64/12.1/src</code>	Fortran source files for included modules.
<code>/opt/pgi/osx86-64/12.1/mpi</code>	osx86-64 OpenMPI for local development

End-user Environment Settings on MacOS

Once software installation is complete, whenever you bring up a PGI command window by double-clicking on the `PGI2012.terminal` file, PGI compilers and tools are accessible and your environment is pre-initialized. You can, however, use the following information to initialize your Terminal shell manually for use of the PGI products.

Each of the following commands assume this:

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

Make 32-bit compilers the default

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/osx86/12.1/man; export MANPATH
$ LM_LICENSE_FILE=$PGI/license.dat; export LM_LICENSE_FILE
$ PATH=$PGI/osx86/12.1/bin:$PATH; export PATH
```

In `csh`, use these commands:

```
% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/osx86/12.1/man
% setenv LM_LICENSE_FILE $PGI/license.dat
% set path = ($PGI/osx86/12.1/bin $path)
```

Make 64-bit compilers the default

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/osx86-64/12.1/man; export MANPATH
$ LM_LICENSE_FILE=$PGI/license.dat; export LM_LICENSE_FILE
$ PATH=$PGI/osx86-64/12.1/bin:$PATH; export PATH
```

In `csh`, use these commands:

```
% setenv PGI /opt/pgi
% setenv MANPATH "$MANPATH":$PGI/osx86-64/12.1/man
% setenv LM_LICENSE_FILE $PGI/license.dat
% set path = ($PGI/osx86-64/12.1/bin $path)
```

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

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

```
$ PATH=$PGI/osx86-64/12.1/mpi/openmpi/bin:$PATH
$ export PATH
```

In `csh`, use this commands:

```
% set path = ($PGI/osx86-64/12.1/mpi/openmpi/bin $path)
```


Chapter 6. Installations on Microsoft Windows

This section describes how to install *PGI Workstation* on a system running a Microsoft Windows operating system. These instructions are valid for both 64-bit and 32-bit 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

PGI products for Windows include 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 include a subset of the full Microsoft Platform SDK. PGI products 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>.

Note

Verify you have enough free disk space. PGI software requires 750 MB of free disk space during the installation.

Installation Steps for Windows

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

Note

Administrator privileges are required to install PGI software.

1. Log in as Administrator.

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

2. Start the installation.

Note

Administrator privileges are required to install PGI software.

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

pgiws-120.exe	32-bit Windows
pgiwsx64-120.exe	64-bit/32-bit Windows

License Setup Tool:

The installer runs the PGI Licensing Setup tool at the conclusion of the installation 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:

- o Be connected to the Internet.
- o Have Microsoft .NET Framework 2.0 or Microsoft .NET Framework 3.5 installed.
- o 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.

Important

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 generation and installation process. It prompts you for your PGI username and password as well as the type of license you wish to install, such as a 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.

3. Make PGI products accessible and prepare for licensing.

Once software installation is complete, 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 is pre-initialized.

4. Verify release number.

Verify the release number of the products you have installed. Open *PGI Workstation* from your desktop by double-left-clicking the PGI icon and entering one of the following commands. The release number is in the first line displayed in the BASH shell window.

For Fortran 77, use: **pgf77 -V x.f**
 For Fortran 95, use: **pgfortran -V x.f**
 For C++, use: **pgCC -V x.cc**
 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. Use it to 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 e-mail from `accounts-noreply@pgroup.com`.
- If you don't have an account, you can create one at: `www.pgroup.com/register`.
- The FlexNet hostid and hostname of the computer on which the software is installed. The installer echoes this information to your screen.

Note

After you have installed the PGI software, you can obtain your FlexNet hostid by double-left-clicking on the *PGI Workstation* desktop icon to open a *PGI Workstation* command 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 e-mail and password in your order acknowledgment.

```
FlexNet Host ID: 001234A98765
Installation: C:\Program Files\PGI\
PGI Release: 2012
```

You can also obtain evaluation license keys by logging in at www.pgroup.com/login with your existing PGI user account or by creating a new account if you have not previously registered.

Generate License Keys

Log in to your PGI web account at www.pgroup.com/login. You should see a screen similar to the following:

Welcome

Use the links below to manage your PGI account.

[Download software](#) – For updating or evaluation.

[Manage PGI products](#) – Purchase, subscription, and ownership information

[Create permanent keys](#) – Create permanent software license keys

[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.

[Update account](#) – Update contact information, change password, or modify email preferences.

[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 e-mail, in the section labeled: 'SHARING ADMINISTRATION.'
 - 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, 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 previously received license keys from PGI, replace the contents of the `license.dat` file created during installation with the license keys you generated using the preceding steps.
 - If your `license.dat` file already contains keys that you previously received from PGI, first make a copy of the existing `license.dat` file in case you encounter a problem with the new license. Then do one of the following:
 - If the license keys in the file are for a previous release of the same product, overwrite the keys.
 - If the license keys in the file are for different PGI products, append the new keys to the keys already in this file.
6. Start the PGI license server.

Note

If you are evaluating a PGI product using trial keys, skip to Step 7. You do not need to start the license server.

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

The PGI License Server is a Windows Service. To start it, follow these steps:

1. Open the Services dialog from the Start menu:

Start | Control Panel | Administrative Tools | Services
2. Select *PGI License Server*.
3. Select *Start*, if the PGI service is stopped. If the PGI service is running, *STOP* and then *START* it. This process guarantees that the new PGI license is served and that the older, cached PGI license is no longer active.

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 documentation by using any web browser to go to:

www.pgroup.com/resources/docs.htm

Tip

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

8. Customize *PGI Workstation*

Optionally, customize the setup, as described in the next sections, [“Customizing the Command Window”](#) and [“PGI Workstation Directory Structure”](#).

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

Table 6.1. Default Windows Installation Directories by Platform

Platform	Default Installation Directory
32-bit	%SYSTEMDRIVE%\Program Files\PGI\win32\12.1\
64-bit	%SYSTEMDRIVE%\Program Files\PGI\win64\12.1\ %SYSTEMDRIVE%\Program Files (x86)\PGI\win32\12.1\

In addition to the product directories, the Microsoft Open Tools and, optionally, **cygwin** are installed here.

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

In release 11.2, the version of Cygwin shipped with PGI Workstation/Server on Windows was upgraded to Cygwin 1.7.7-1. This is the current version of Cygwin as of February, 2011. In general, PGI users should not notice significant differences in Cygwin behavior. However, there have been some changes from the previous version of Cygwin. For more information about the changes, visit the PGI User Forum topic:

<http://www.pgroup.com/userforum/viewtopic.php?t=2448&sid=6bba0fa561c67762f5e26353592b5303>

For information about configuring and troubleshooting Cygwin, you can visit the following websites:

Cygwin FAQ: <http://cygwin.com/faq.html>

Cygwin Users Guide: <http://cygwin.com/cygwin-ug-net.html>

Cygwin mailing lists and archives: <http://cygwin.com/lists.html>

PGI Workstation Directories on a 64-bit Windows System

The following directory structure is created during installation on a 64-bit system; here *c* is assumed to be the system drive.

Table 6.2. 64-bit Windows Directory Structure

Name of directory	Contents
C:\Program Files\PGI\win64\12.1\bin C:\Program Files (x86)\PGI\win32\12.1\bin	64-bit compilers and tools binaries
C:\Program Files\PGI\win64\12.1\lib C:\Program Files (x86)\PGI\win32\12.1\lib	64-bit runtime and support libraries
C:\Program Files\PGI\win64\12.1\include C:\Program Files (x86)\PGI\win32\12.1\include	64-bit header files
C:\Program Files\PGI\win64\12.1\REDIST C:\Program Files (x86)\PGI\win32\12.1\REDIST	Re-distributable runtime libraries
C:\Program Files\PGI\win64\12.1\doc C:\Program Files (x86)\PGI\win32\12.1\doc	Documentation
C:\Program Files\PGI\win64\12.1\man C:\Program Files (x86)\PGI\win32\12.1\man	Man pages for commands
C:\Program Files\PGI\Microsoft Open Tools 9	Microsoft tools
C:\cygwin	Cygwin tools

PGI Workstation Directories on a 32-bit Windows System

The following directory structure is created during installation on a 32-bit system. Here *c* is assumed to be the system drive.

Table 6.3. 32-bit Windows Directory Structure

Name of directory	Contents
C:\Program Files\PGI\win32\12.1\bin	32-bit compilers and tools binaries
C:\Program Files\PGI\win32\12.1\lib	32-bit runtime and support libraries
C:\Program Files\PGI\win32\12.1\include	32-bit header files
C:\Program Files (x86)\PGI\win32\12.1\REDIST	Re-distributable runtime libraries
C:\Program Files\PGI\win32\12.1\doc	Documentation
C:\Program Files\PGI\win32\12.1\man	Man pages for commands
C:\Program Files\PGI\Microsoft Open Tools 9	Microsoft tools
C:\cygwin	Cygwin tools

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 products 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 keys 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 keys 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 generates 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 the value of 1.

Windows Firewall and PGI Debugging

Microsoft Windows Firewall runs by default on some types of systems, such as Windows XP and Windows Vista. Suppose this application is running on your system. Unless the PGI debugger is included in the Windows Firewall Exceptions list, Windows Firewall may try to block the PGI debugger engine when you debug a program.

Automatic Additions to Windows Firewall Exceptions List

When PGI software is installed

If Windows Firewall is enabled when PGI software is installed, certain components are automatically added to the Exceptions list. These components appear as follows:

PGI Debugger 2012 (32-bit)

PGI Debugger 2012 (64-bit) - not added to the Exceptions list on 32-bit systems.

When PGI software is uninstalled

If Windows Firewall is enabled when PGI software is uninstalled, the components previously listed are automatically removed from the Exceptions list.

When using PGDBG to Debug

If PGDBG is not listed in the Exceptions list and Windows Firewall is enabled when you use PGDBG to debug, the Firewall displays a dialog that allows you to Unblock the application named **pgdebug**. If you choose the Unblock option, Windows Firewall adds the debugger (pgdebug) to its list of Exceptions and you do not see the pop-up dialog again unless you reinstall.

Tip

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

Note

You can manually add these components to the Windows Firewall list of Exceptions.

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 FlexNet 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, you may see one of the following FlexNet License Manager dialog messages: `'LICENSE MANAGER PROBLEM: Cannot connect to license server system.'` or `'LICENSE MANAGER PROBLEM: Failed to checkout license'`

This message may appear due to one of the following:

The PGI License Server is not running.

Firewall problems exist.

The Hostname in the license is not one the license server can find successfully. This hostname must map to the IP address of the server.

- 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:

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

- By default, on Windows, the license server generates 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.

For assistance with difficulties related to the installation, send e-mail to trs@pgroup.com.

Chapter 7. Contact Information

You can contact The Portland Group at:

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

Or 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

Many questions and problems can be resolved by following instructions and the information available at our frequently asked questions (FAQ) site:

www.pgroup.com/support/faq.htm

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

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.

