PGDBG Installation Guide

Version 2015
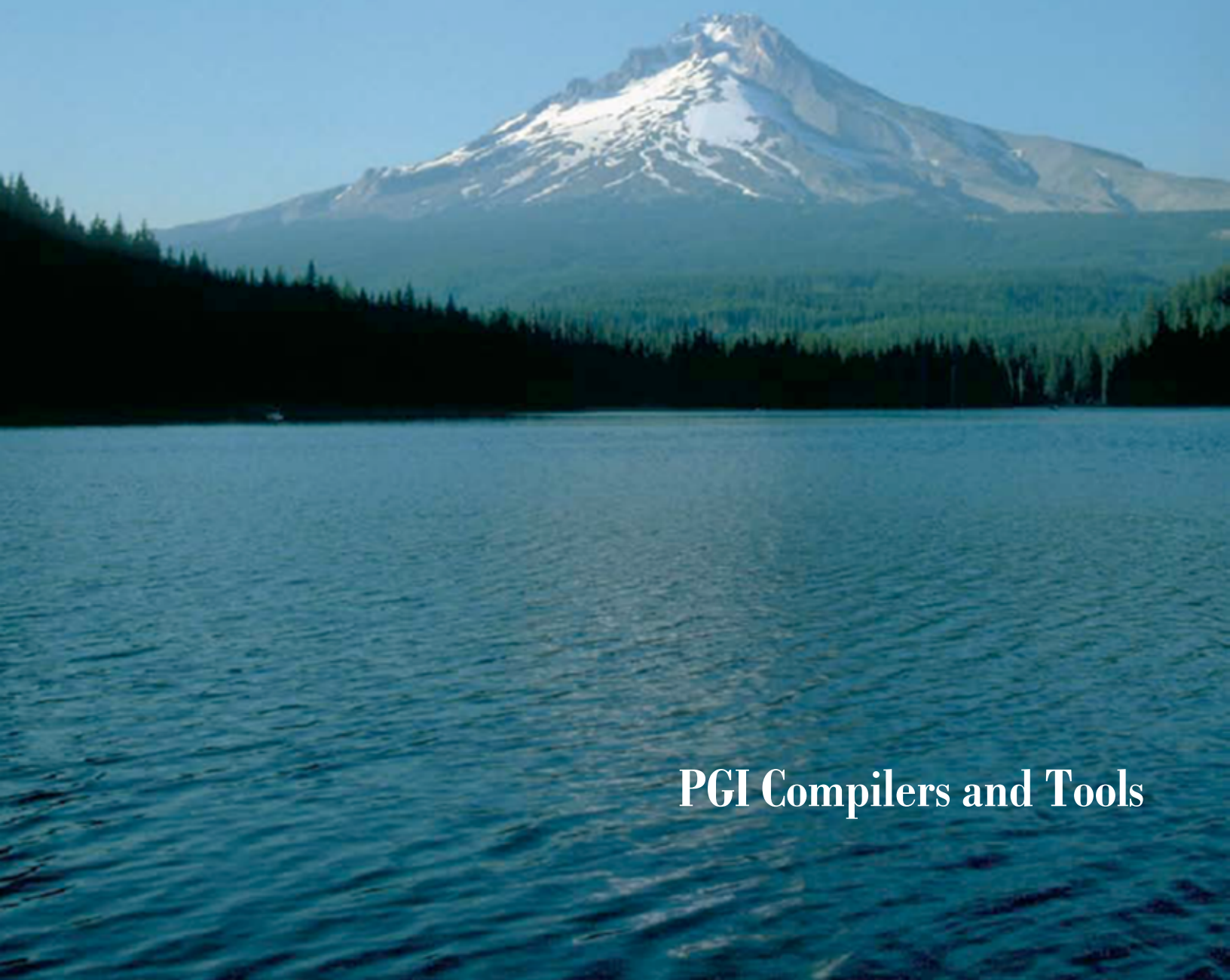
PGI Compilers and Tools

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1.
RELEASE 2015 INTRODUCTION

Welcome to Release 2015 of the PGDBG® for 32-bit and 64-bit x86-compatible processor-based workstations, servers, and clusters running versions of the Linux, Apple OS X, and Microsoft Windows operating systems. This installation information applies to the PGDBG installation package available from The Portland Group.

1.1. Product Overview

PGDBG is licensed software available from The Portland Group. You must agree to an End-User License Agreement when you install the software.

PGDBG supports debugging programs running on local and remote systems. The PGI license keys that enable PGDBG to debug must be located on the same system where the program you want to debug is running.

Local debugging
  If you want to debug a program running on the system where you have launched PGDBG, you are doing local debugging and you need license keys on that local system.

Remote debugging
  If you want to debug a program running on a system other than the one on which PGDBG is launched, you are doing remote debugging and you need license keys on the remote system. The remote system also needs an installed copy of PGI Workstation, PGI Server, or PGI CDK.

1.1.1. PGI Remote and PGI Local Debugging Comparison

The PGDBG install package includes the PGDBG debugger. It supports debugging applications locally as well as on a system other than the one on which PGDBG is installed.

Figure 1 illustrates local debugging and its license key requirements.
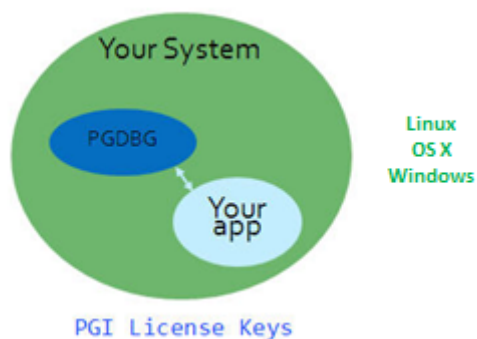
Figure 1   Local Debugging Licensing

Figure 2 illustrates remote debugging and its license key requirements.
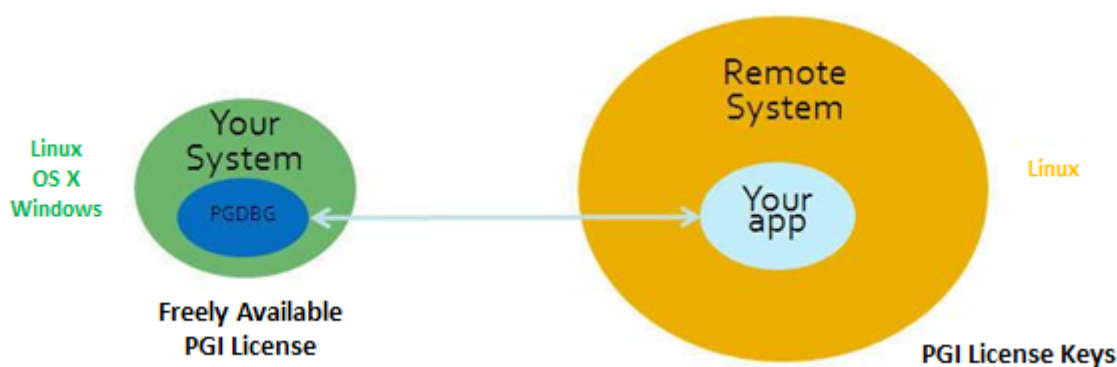
Figure 2   Remote Debugging Licensing

1.2. Release Components

Release 2015 of PGDBG includes PGI's multi-thread graphical debugger.

1.2.1. Co-installation Requirements

If the program to be debugged is running on the same system as PGDBG all you need to debug locally are valid license keys.

If the program to be debugged is running on a different system than PGDBG, then either PGI Workstation, PGI Server, or PGI CDK must be installed on that remote system and valid license keys must be in placed there. The remote system must be running a Linux operating system. The license required to launch PGDBG on your local system (before beginning a remote debug session) is freely available from PGI.

1.3. Supported Operating Systems

PGDBG supports Linux, Windows, and Apple operating systems:

▶ Linux

Most Linux operating systems with GLIBC 2.3.4 and above:

> CentOS 5.2+, including CentOS 7
> Fedora 4+, including Fedora 21
> RHEL 4.8+, including RHEL 7
> SLES 10+, including SLES 11 SP 3
> SuSE 9.3+, including openSuSE 13.2
> Ubuntu 8.04+, including Ubuntu 14.10

Version differences in some operating system components may cause difficulties, but often these can be overcome with minor adjustments to the PGI software installation or operating system environment.

▶ OS X

Supported versions are OS X versions 10.7 (Lion) and newer, including 10.10 (Yosemite).

▶ Windows

> Windows Server 2008 R2
> Windows 7
> Windows 8
> Windows 8.1
> Windows Server 2012

> www.pgroup.com/support/install.htm lists any new operating system distributions that may be explicitly supported by PGDBG. If your operating system is newer than any of the operating systems in the preceding list, the installation may still be successful.

Linux

▶ Linux operating systems with support for x64 compatible processors are designated 64-bit. 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.

Most modern operating systems include support for Intel Hyper-threading (HT). Further, most modern Linux distributions support the *Native Posix Threads Library (NPTL)*. 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.

Windows

PGI 2015 for Windows supports most of the features of the 32-bit and 64-bit versions for Linux environments.

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

OS X

PGI 2015 for OS X supports most of the features of the 32-bit and 64-bit versions for Linux environments. Except where noted in PGI documentation, the PGI compilers and tools on OS X function identically to their Linux counterparts.

1.4. Java Runtime Environment (JRE)

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

1.5. Product Support

All new PGI licenses include free PGI Subscription Service during the money-back guarantee period.

The PGI Subscription Service provides support and other benefits including:

▸ Ongoing technical support by electronic mail. 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.
▸ Release upgrades for licensed product(s) at no additional cost, except for any administrative fee that may apply.
▸ Notification by email when maintenance releases occur and are available for electronic download and installation.
▸ 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 the money-back guarantee period days, you can purchase an extension to your PGI Service Subscription. 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.
INSTALLATION OVERVIEW

This section provides an overview of the steps required to successfully install PGDBG. The remaining sections provide the details of each of the steps. Licensing was covered in Release 2015 Introduction. PGDBG Installations on Linux describes how to install PGDBG in a generic manner on Linux, Installations on OS X describes how to install on an OS X system, and Installations on Microsoft Windows describes how to install on a Windows system.

2.1. Before You Begin

Before you begin the installation, it is advantageous to understand the flow of the installation process. There are two 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.

The following illustration provides a high-level overview of the PGI installation process.
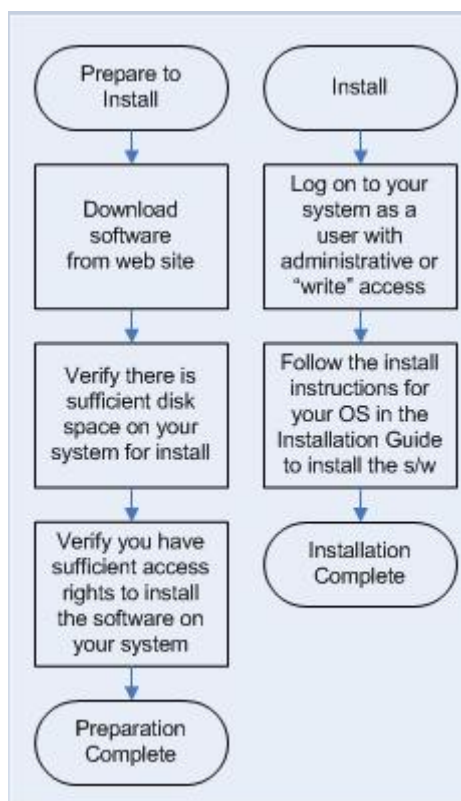
Figure 3   PGDBG Installation Overview

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

2.2. PGI Network Installations

Linux

On Linux, PGDBG 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 installation. For more information on network installations, refer to PGDBG Installations on Linux.

OS X and Windows

On OS X and Windows, PGDBG must be installed locally on each machine. A network installation using a shared file system is not supported.

Chapter 3.
PGDBG INSTALLATIONS ON LINUX

This section describes how to install PGDBG in a generic manner on a Linux system. It covers local and network installs.

▶ For installations on 32-bit x86 systems, the PGI installation script installs only the linux86 versions of PGDBG.

▶ 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 PGDBG.

▶ If the 32-bit gcc development package is already installed on the x64 system, the 32-bit linux86 PGDBG is also installed.

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

3.1. Prepare to Install PGDBG on Linux

In preparation for installing PGDBG on Linux, first study the overview of the Linux installation process illustrated in Figure 4.

> 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`.
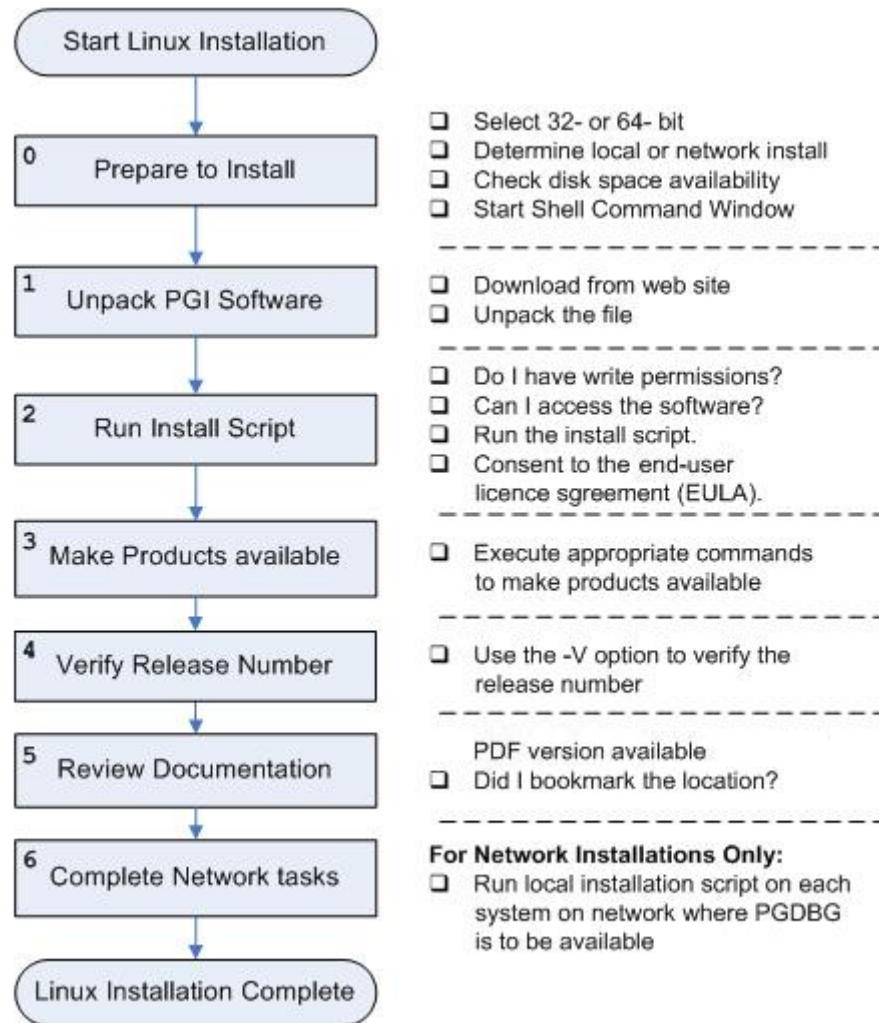
Figure 4 Linux Installation Overview

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 PGDBG will be used. This directory selection approach allows a network installation to support a network of machines running different versions of Linux.

To Prepare for the Installation:

▸ 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 70 MB of free disk space.
  - ▸ The linux86 platform requires up to 77 MB of free disk space.
  - ▸ The linux86-64 platform requires up to 77 MB of free disk space.

- ▸ 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`.

3.2. PGI Installations on Linux - gcc Compiler Requirements

On Linux systems, PGDBG needs 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 on a Linux system, the system must already have gcc compilers installed.

- ▸ To successfully install the 64-bit PGDBG, gcc 64-bit compilers must be present and in the user's path.
- ▸ To successfully install the 32-bit PGDBG, gcc 32-bit compilers must be present and in the user's path.

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

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

3.2.1. 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?
   ```

   Output should look similar to the following:
   ```
   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?
   ```

Output should look similar to the following:

```
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
```

3.3. Installation Steps for Linux

Follow these instructions to install the software.

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

2. Unpack the PGI software.

   In the instructions that follow, replace `<tarfile>` with the name of the file that you downloaded.

   > The PGI products cannot be installed into the same directory where the tar file is unpacked.
   >
   > If you are installing both 64-bit and 32-bit products, unpack each tar file into a separate temporary directory.

   Use the following command sequence to unpack the tar file into a temporary directory before installation.

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

3. Run the installation script.

   > The installation script for PGDBG is **install**. These instructions refer to this as the installation script.

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

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

   ```
   % ./install
   ```

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

   ▶ Consent to the End-User License Agreement (EULA).
   ▶ Determine whether to perform a local installation or a network installation.
   ▶ Define where to place the installation directory.

   After the software is installed, the installation script performs system-specific customization.

   If you are performing a network installation for PGDBG, 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.

4. Make PGDBG accessible.

After software installation is complete, each user's shell environment must be initialized to use PGDBG.

64-bit as default

To make the 64-bit PGDBG the default:

In csh, use these commands:

```
% setenv PGI /opt/pgi
% set path = ($PGI/linux86-64/15.1/bin $path)
```

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

```
$ PGI=/opt/pgi; export PGI
$ PATH=$PGI/linux86-64/15.1/bin:$PATH; export PATH
```

32-bit as default

To make the 32-bit PGDBG the default:

In csh, use these commands:

```
% setenv PGI /opt/pgi
% set path = ($PGI/linux86/15.1/bin $path)
```

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

```
$ PGI=/opt/pgi; export PGI
$ PATH=$PGI/linux86/15.1/bin:$PATH; export PATH
```

> PGDBG supports debugging both 32-bit and 64-bit programs. PGDBG automatically determines the architecture of the program and configures itself accordingly.

5. Verify the release number of the installed software.

To verify the release number of the products you have installed, use the -V option.

```
pgdbg -V
```

6. Review documentation.

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

```
$PGI/linux86-64/15.1/doc/index.htm
```
or
```
$PGI/linux86/15.1/doc/index.htm
```

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

7. Complete network installation tasks. [Network Installations only]

For a network installation, you must run the local installation script on each system on the network where PGDBG will be available for use.

If your installation base directory is /opt/pgi and /usr/pgi/shared/15.1 is the common local directory, then run the following commands on each system on the network.

```
/opt/pgi/linux86/15.1/bin/makelocalrc  -x /opt/pgi/linux86/15.1 \
    -net /usr/pgi/shared/15.1
/opt/pgi/linux86-64/15.1/bin/makelocalrc  -x /opt/pgi/linux86-64/15.1 \
    -net /usr/pgi/shared/15.1
```

These commands create a system-dependent file `localrc.machinename` in both of these directories: `/opt/pgi/linux86/15.1/bin` and `/opt/pgi/linux86-64/15.1/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/15.1/lib
/usr/pgi/shared/15.1/liblf
/usr/pgi/shared/15.1/lib64
```

> 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 PGDBG for Linux is now complete.

Chapter 4.
INSTALLATIONS ON OS X

This section describes how to install PGDBG on a computer running Apple OS X.

> PGI products for Apple OS X are supported on Intel processor-based Macintosh computers running OS X version 10.6.0 or greater with Apple Xcode 3.2 or greater installed. Xcode is available from http://developer.apple.com.

The default installation base directory is `/opt/pgi`.

▶ 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 a 64-bit system, the 32-bit osx86 tools are also installed.

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

4.1. System Configuration to Enable Debugger 'Attach'

The PGDBG debugger can debug programs that are launched by the debugger, either by specifying the program on the PGDBG command line, using the `debug` or `load` commands, or by specifying a program in the Connections tab of the PGDBG GUI.

OS X security features prevent PGDBG from attaching to a running program unless some configuration is performed. Users who need to attach PGDBG to a running program must be able to enter the authentication credentials for a user that is a member of the _developer group. Users can be added to a group using the following command:

```
sudo dseditgroup -o edit -a usernametoadd -t user _developer
```

4.2. Installation Steps for Apple OS X

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

1. Access the installation package.

   Download the software from www.pgroup.com or another electronic distribution site. The file you download is `PGI_Debugger_144.dmg`. If it does not mount automatically, open this file to mount it.

2. Install the software.

   Double-click the `PGI Debugger 15.1.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 3.2 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.

3. Make PGI products accessible.

   To initialize the PGI compilers and tools, import the `PGI 2015` terminal settings file located in the PGI installation directory using Terminal's Shell | Import command. You can access the installation directory in the Open file dialog by typing `Command-Shift-G` and entering the path, for example `/opt/pgi`.

   Once initialized, you can open a Terminal session preconfigured with the PGI environment directly from within Terminal. To do this, bring up a PGI window by using either the `Shell | New Window` or `Shell | New Tab` menu commands; and then select `PGI 2015`.

   For more information on configuring to make either 32-bit or 64-bit compilers the default, refer to End-user Environment Settings on OS X.

4. Verify the release number.

   To verify the release number, use the `-V` option with the debugger command:

   ```
   pgdbg -V
   ```

5. Review documentation.

   The documentation is in PDF format and located in:

   ```
   /opt/pgi/osx86/15.1/doc/
   or
   /opt/pgi/osx86-64/15.1/doc/
   ```

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

4.3. End-user Environment Settings on OS X

Once software installation is complete, and PGI Terminal settings are imported, whenever you bring up a PGI command window from within Terminal by choosing the `PGI 2015` terminal option, PGDBG is accessible and your environment is pre-initialized. If you prefer to manually initialize your Terminal shell for use of the PGI products, use the following information.

Make 32-bit PGDBG the default

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

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

```
$ export PATH=/opt/pgi/osx86/15.1/bin:$PATH
```

In csh, use this commands:

```
% setenv PATH /opt/pgi/osx86/15.1/bin:$PATH
```

Make 64-bit PGDBG the default

The following commands make the 64-bit PGDBG the default.

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

```
$ export PATH=/opt/pgi/osx86-64/15.1/bin:$PATH
```

In csh, use these commands:

```
% setenv PATH /opt/pgi/osx86-64/15.1/bin:$PATH
```

PGDBG supports debugging both 32-bit and 64-bit programs. PGDBG automatically determines the architecture of the program and configures itself accordingly.

Chapter 5.
INSTALLATIONS ON MICROSOFT WINDOWS

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

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 PGDBG. For installations on 32-bit x86 systems, the PGI installer installs only the 32-bit versions of PGDBG.

5.1. Preparing to Install on Windows

Verify you have enough free disk space. PGDBG requires 150 MB of free disk space during the installation.

5.2. Installation Steps for Windows

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

> 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.

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

pgdbg32-144.exe	32-bit Windows
pgdbgx64-144.exe	64-bit Windows

3. Review documentation.

    You can view the online documentation which is included with your installation and is also available by using any web browser and going to www.pgroup.com/resources/docs.htm

5.3. Windows Firewall and PGI Debugging

Microsoft Windows Firewall runs by default on some Windows systems. 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.

5.3.1. 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 2015 (32-bit)
> PGI Debugger 2015 (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.

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.

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

Chapter 6.
CONTACT INFORMATION

You can contact PGI at:

20400 NW Amberwood Drive Suite 100
Beaverton, OR 97006

Or electronically using any of the following means:

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

http://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:

http://www.pgroup.com/support/faq.htm

All technical support is by e-mail or submissions using an online form at:

http://www.pgroup.com/support

Phone support is not currently available.

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

PGI®