



PathScale™ Compiler Suite and Subscription Manager Install Guide

Version 3.2

Information furnished in this manual is believed to be accurate and reliable. However, PathScale LLC assumes no responsibility for its use, nor for any infringements of patents or other rights of third parties which may result from its use. PathScale LLC reserves the right to change product specifications at any time without notice. Applications described in this document for any of these products are for illustrative purposes only. PathScale LLC makes no representation nor warranty that such applications are suitable for the specified use without further testing or modification. PathScale LLC assumes no responsibility for any errors that may appear in this document.

No part of this document may be copied nor reproduced by any means, nor translated nor transmitted to any magnetic medium without the express written consent of PathScale LLC. In accordance with the terms of their valid PathScale agreements, customers are permitted to make electronic and paper copies of this document for their own exclusive use.

Linux is a registered trademark of Linus Torvalds.

PathScale, the PathScale logo, and EKOPath are registered trademarks of PathScale, LLC.

Red Hat and all Red Hat-based trademarks are trademarks or registered trademarks of Red Hat, Inc.

SuSE is a registered trademark of SuSE Linux AG.

All other brand and product names are trademarks or registered trademarks of their respective owners.

© 2007 PathScale, LLC. All rights reserved.
© 2006, 2007 QLogic Corporation. All rights reserved worldwide.
© PathScale 2004, 2005, 2006. All rights reserved.
First Published: April 2004
Printed in U.S.A.

PathScale LLC, 2071 Stierlin Ct., Suite 200, Mountain View, CA 94043

Table of Contents

Section 1

Introduction

- 1.1 Conventions used in this document 1-2
- 1.2 Documentation Suite 1-2

Section 2

Installation Prerequisites

- 2.1 System and Linux Distribution Requirements. 2-1
- 2.2 Compiler Suite Software 2-2
- 2.3 Subscription File 2-2

Section 3

Installing the PathScale Compiler Suite

- 3.1 Linux Distribution 3-1
- 3.2 Compiler Software Distribution for Your Linux Distribution. 3-2
 - 3.2.1 Installation Method and Media 3-2
- 3.3 Subscription Management 3-3
 - 3.3.1 Types of Subscriptions 3-3
 - 3.3.2 Obtain the Subscription File 3-4
 - 3.3.3 Get the Hostname, IP Address, and MAC Address 3-4
 - 3.3.4 Send the Information 3-5
 - 3.3.5 Installing the Subscription File 3-6
- 3.4 Download Software. 3-6

3.5	Typical Install Scenarios and Terminology	3-6
3.6	Migrating from Previous Releases	3-7
3.7	Accessing the ISO Image or Physical CD	3-8
3.7.1	Mount the ISO Image	3-8
3.7.2	Accessing the Physical CD	3-8
3.8	Installing the Full Development Environment Using RPMs	3-9
3.8.1	Extract the RPMs from the Download File	3-9
3.8.2	Installing the Compiler RPMs	3-9
3.8.3	Installing for a 32-bit platform	3-10
3.8.4	Changing the Default Installation Directory	3-10
3.8.5	Installing Multiple RPM Versions	3-11
3.8.6	Options for Multiple RPM Installs	3-11
3.8.7	Files Modified by RPM Install	3-12
3.8.8	Verify Subscription File	3-13
3.9	Installing the Full Development Environment Using DEBs	3-13
3.9.1	Extract the DEB from the Download File	3-13
3.9.2	Installing the Compiler DEB	3-14
3.9.3	Files Modified by DEB Install	3-16
3.9.4	Verify Subscription File	3-16
3.10	Installing the Full Development Environment as a Non-root User	3-16
3.10.1	Extract the Non-root tar File	3-17

3.10.2	Installing the Compilers from a Non-root tar File	3-17
3.10.3	Installing Multiple Non-root tar File Versions	3-18
3.11	Installing Subsets of the Compiler Suite	3-18
3.11.1	After Installation	3-18
3.12	Setting Environment Variables	3-19
3.12.1	Using Shared Libraries	3-19
3.12.2	Setting the PATH	3-20
3.12.3	Access to Compiler <code>man</code> Pages with <code>MANPATH</code>	3-20
3.12.4	Setting Up the Compilers for Use Over NFS	3-21
3.12.5	Setting and Using <code>NLSPATH</code>	3-22
3.13	PathScale Subscription Management	3-22
3.14	Installing the Subscription Manager Client	3-23
3.15	About Floating Subscriptions	3-23
3.16	Installing the Subscription Manager Server for Floating Mode	3-24
3.17	Configuring the Environment for Floating Subscriptions	3-24
3.17.1	Environment Variables for Subscription Management	3-24
3.17.2	The Subscription Daemon	3-25
3.17.3	Proxies	3-26
3.17.4	Running the Daemon on a Different Interface	3-26
3.17.5	Running the Subscription Server	3-27
3.18	The Subscription File	3-27

3.18.1	Subscription File for Nodelocked Subscriptions	3-28
3.18.2	Subscription File for Floating Subscriptions	3-28
3.19	Upgrading an Existing Installation	3-29
3.20	Monitoring Lease Usage	3-30
3.21	Verifying and Running the Compiler	3-31
3.21.1	Hello World	3-32

Section 4

Installing Subsets of the PathScale Compiler Suite

4.0.1	Installing Individual Compilers and Runtime Environment	4-1
4.0.2	Installing the PathScale Debugger and Runtime Environment	4-2
4.0.2.1	Installing PathDB Only	4-2
4.0.3	Installing the Runtime Environment Only	4-3

Section 5

PathScale Installer

5.1	Where to Find the Installer	5-1
5.2	What You Need to Run the Installer	5-1
5.3	Downloading Compiler Suite Software	5-2
5.3.1	Unpack tar File (RPM and Non-root tar Install Only)	5-2
5.4	Mounting an ISO Image	5-2
5.5	Accessing the Physical CD	5-3
5.6	Running the Installer	5-3
5.7	What the Installer is Doing	5-4

5.8	Exiting the Installer	5-4
5.9	Use the Installer to Upgrade or Uninstall Compilers	5-4
Section 6		
Uninstalling the Compiler		
6.1	Uninstalling compiler RPMs	6-1
6.1.1	Uninstalling One of Multiple Compiler Versions of RPMs	6-1
6.2	Uninstalling compiler using Debian Packages	6-2
6.3	sUninstalling Compilers from a Non-root tar File	6-2
6.3.1	Uninstalling One of Multiple Compiler Versions of Non-root tar Files	6-2
Appendix A		
Subscription Management Details		
A.1	Modes of Operation	A-1
A.2	Nodelocked Single User	A-1
A.3	Nodelocked, Multiple User	A-2
A.4	Floating	A-2
A.4.1	Named Users for Floating Subscription	A-3
A.5	Subscription File	A-3
A.6	Perpetual Right-to-use	A-4
A.7	Overview of Subscription Manager Client-Server Interaction	A-4
Appendix B		
Troubleshooting		
B.1	Installation Problems	B-1
B.1.1	Problem. Trouble Installing on SLES or SUSE 9.x	B-1

B.1.2	Problem: Installing SUSE/SLES, Non-default Location	B-1
B.1.3	Problem: Failed Dependencies	B-2
B.1.4	Problem: Missing Packages	B-2
B.1.5	Problem: Removing Dependent Packages	B-2
B.2	Subscription Problems	B-3
B.2.1	Can't Find Subscription File	B-3
B.2.2	Misconfigured Environment	B-4
B.2.3	Getting Information on Subscription Problems	B-4
B.2.4	Problems Starting the Subscription Server	B-4
B.2.5	Server is Running But the Compiler Can't Find It	B-5
B.2.6	Server is Not Running	B-7
B.2.7	Problem: Server Fails to Start	B-7
B.2.8	Problem: Compiler says "Unable to obtain subscription" and then Stops	B-8
B.2.9	Problem: Compiler: "Bad subscription version: <text>" and then Stops	B-8
B.2.10	Problem: Compiler: "No subscription file found" and then stops	B-8
B.2.11	Problem: Unable to Upgrade with Current Subscription:	B-9
B.2.12	Problem: Compiler prints lines of the form: *** Subscription: <text>	B-9
B.2.13	Problem: Ethernet Port is Down	B-9
B.2.14	Problem: Nodelocked Perpetual Subscription Fails	B-10
B.2.15	Problem: Floating Perpetual Subscription Fails	B-10
B.2.16	Problem: Subscription Server Hangs with ROCKS	B-11

B.3	Other Problems	B-11
B.3.1	NLSPATH Causes Error	B-11

Appendix C

Glossary

Appendix D

Package Dependencies

D.1	Specific Packages by Platform s	D-1
D.1.1	Red Hat Enterprise Linux Workstation 4	D-2
D.1.2	Red Hat Enterprise Linux Workstation 5	D-4
D.1.3	Fedora Core 3	D-6
D.1.4	Fedora Core 4	D-8
D.1.5	Fedora Core 5	D-10
D.1.6	Fedora Core 6	D-12
D.1.7	Fedora Core 7	D-14
D.1.8	Fedora Core 8	D-16
D.1.9	SUSE Linux Enterprise Server (SLES) 9	D-18
D.1.10	SUSE Linux Enterprise 10	D-20
D.1.11	SUSE Linux Professional 9.3	D-22
D.1.12	SUSE Linux Professional 10.0	D-24
D.1.13	SUSE Linux Professional 10.3	D-26

Figures

Figure		Page
3-1	Example of Lease Usage Page	3-27

5-1	Installer Welcome Screen	5-4
-----	------------------------------------	-----

Tables

Table		Page
2-1	Supported Linux Distributions and Platforms	2-1
3-1	Compiler Distributions for Different Linux Environments.	3-1
3-2	Installation Media Contents	3-2
3-3	Types of Installation	3-6

Section 1

Introduction

This document explains how to install and run the PathScale™ Compiler Suite and Subscription Manager. The Compiler Suite includes optimizing compilers and runtime support for C, C++, and Fortran. The C and C++ compilers are based on GNU GCC 3.x and 4.x front-ends. See the *PathScale Compiler Suite User Guide* for more information.

The Fortran compilers are based on the SGI/Cray Open64 Fortran compiler and these have been extended to increase compatibility with the GNU `g77` Fortran compiler. These compilers produce code that follows the Linux `x86_64` ABI and run on both the AMD64® family of chips and the Intel® EM64T family of chips.

The PathScale Compiler Suite will be referred to as the PathScale Compiler Suite, the Compiler Suite, or the compiler for the rest of this document.

This means that object files produced by the PathScale compilers can link with object files produced by other Linux `x86_64`-compliant compilers such as Red Hat and SUSE GNU `gcc`, `g++`, and `g77`. The AMD64 platform is the AMD 64-bit extension to the Intel IA32 architecture, often referred to as “x86”. EM64T is the Intel *Extended Memory 64 Technology* chip family.

Depending on the type of subscription you purchased, you enabled some or all of the following:

- PathScale C Compiler for `x86_64` architecture
- PathScale C++ Compiler for `x86_64` architecture
- PathScale Fortran Compiler for `x86_64` architecture
- Documentation
- Libraries
- Subscription Manager client. You must have a valid subscription and associated subscription file in order to run the compiler.
- Subscription Manager server (optional). The PathScale Subscription Manager server is required for floating subscriptions.
- PathScale Debugger (`pathdb`). See the *PathScale Debugger User Guide* for more information on the debugger.

For information on optimizing the compilers, see the *PathScale Compiler Suite User Guide*.

1.1 Conventions used in this document

These conventions are used throughout the PathScale documentation.

Convention	Meaning
<code>command</code>	Fixed-space font is used for literal items such as commands, files, routines, and pathnames.
<i>variable</i>	Italic typeface is used for variable names or concepts being defined.
user input	Bold, fixed-space font is used for literal items the user types in. Output is shown in non-bold, fixed-space font.
\$	Indicates a command line prompt
#	Command line prompt as root
[]	Brackets enclose optional portions of a command or directive line.
...	Ellipses indicate that a preceding element can be repeated.
NOTE:	Indicates important information

1.2 Documentation Suite

The PathScale Compiler Suite product documentation set includes:

- *The PathScale Compiler Suite and Subscription Manager Install Guide*
- *The PathScale Compiler Suite User Guide*
- *The PathScale Compiler Suite Support Guide*
- *The PathScale Debugger User Guide*

There are also online manual pages (“man pages”) available describing the flags and options for the PathScale Compiler Suite. These man pages are shipped with the Compiler Suite: `eko`, `pathf95`, `pathcc`, `pathCC`. The `pathscale-info` man page gives an overview of all the various man pages that are included with the Compiler Suite.

Please see the PathScale web site for further information about current releases and product support.

<http://www.pathscale.com/>

The Release Notes (`release_notes.txt`) contain the last minute and late breaking news concerning the compiler release. The PathScale debugger release notes (`release_notes_pathdb.txt`) contain information about PathDB.

Section 2

Installation Prerequisites

Before installing and using the compiler, make sure that you have the following:

- A supported system and Linux distribution
- PathScale Compiler Suite software
- A Subscription File obtained from PathScale

Details on the prerequisites are given below. Instructions for software installation are given in [Section 3](#).

2.1

System and Linux Distribution Requirements

The PathScale Compiler Suite has been tested on systems based on the AMD64 and EM64T families of processors, including Opteron, Athlon64, Athlon64 FX, 32-bit Athlon, Xeon, and Pentium. This version of the PathScale Compiler Suite for Linux supports the Linux distributions listed in [Table 2-1](#).

The Compiler Suite has not been tested on systems with less than 512 megabytes of memory. At least 2GB of RAM is recommended for best performance.

Table 2-1. Supported Linux Distributions and Platforms

Linux Distribution	Platform
Red Hat Enterprise Linux 4, 5 (RHEL4, RHEL5) Fedora Core 3, 4, 5, 6 (FC3, FC4, FC5, FC6) Fedora 7, 8 openSUSE 10.3 SUSE Linux Enterprise 10 ^a SUSE Linux Enterprise Server (SLES) 9 SUSE Linux Professional 9.3, 10, 10.1 Ubuntu 7, 8	AMD64, EM64T, x86 (32-bit)

Table Notes

^aSUSE Linux Enterprise 10 refers to both the Desktop and Server editions.

See the PathScale support website for information on the package dependencies for different platforms.

2.2

Compiler Suite Software

The PathScale Compiler Suite is available in these formats:

- Downloaded as a `tar` file of binary RPMs or Debian `pkg` for a specific Linux distribution from the PathScale web site
- Downloaded as a non-root `tar` file, containing a non-root tarball installer for your specific platform. (This packaging is currently not available for Ubuntu.)
- As a CD image, either by downloading the ISO image or as a physical CD - this contains the binary RPMs, non-root `tar` file, and source `tar` file

The compiler software can be downloaded from the PathScale web site:

<http://www.pathscale.com>

A user ID and password will be required. See [See section 3](#) for more details on downloading and installation.

2.3

Subscription File

The PathScale Compiler Suite and PathDB debugger will not run without a valid and correctly installed subscription file. This is true for both *node-locked* and *floating subscriptions*. The subscription management software includes components that verify your subscription and help you to track your usage of the PathScale software and tools.

When you order a subscription from PathScale, you will be asked to provide information about the systems on which the software you want to use will run. When the requested information has been received, PathScale will send you a file via email that provides the subscription software with the information needed to enable the PathScale products.

Once installed, this information will be used by the subscription management software to verify that the subscription is valid.

[See section 3.3](#) for more details.

Section 3

Installing the PathScale Compiler Suite

This section provides instructions for the installation of the PathScale Compiler Suite. The steps are summarized here:

- Make sure that you are running on a supported platform and Linux distribution
- Determine which software is appropriate for your distribution, and what type of install method you will be using. Then download the appropriate tar file or ISO image containing the software.
- Determine the type of subscription you will need
- Obtain a Subscription File from PathScale
- Install software
- Set environment variables, if necessary
- Install the appropriate subscription management, if using a floating license
- Install Subscription File
- Verify and test the compiler

Details are provided in the following sections.

3.1

Linux Distribution

The tested Linux distributions and platforms are given in [Table 2-1](#) in “Installation Prerequisites” on page 1.

3.2

Compiler Software Distribution for Your Linux Distribution

It is important that the right Compiler Suite distribution is selected for the Linux distribution you are using. The Compiler Suite distributions and their corresponding Linux distributions are displayed in [Table 3-1](#).

Table 3-1. Compiler Distributions for Different Linux Environments

Compiler Distribution Identifier	Used On
Fedora	Red Hat Enterprise Linux Workstation 4 (RHEL4), Red Hat Enterprise Server 5 (RHEL5), Fedora Core 3, Fedora Core 4, Fedora Core 5, Fedora Core 6, Fedora 7
SUSE9	SUSE SLES 9, SUSE SLES 10, SUSE 9.3, openSUSE 10.0, openSUSE 10.1
Ubuntu	Ubuntu 7, Ubuntu 8

New compiler releases will be noted on the and PathScale web site:

<http://www.pathscale.com/support/>

3.2.1

Installation Method and Media

The PathScale Compiler Suite is distributed in these forms. The Linux distributions of the PathScale Compiler Suite are installed from binary RPMs, Debian Packages (DEBs), or a non-root `tar` file. There are slightly different installation procedures for each format.

All except the physical CD can be downloaded from the PathScale web site.

- **Download `tar` file** of binary RPMs or DEBs for a specific Linux distribution. This is referred to as the **download tar file** in this document. Use the RPM or DEB format ([see section 3.8](#)) for easier management and updating of packages and if you have root privileges to install. This method is preferred when using a floating subscription, as the Subscription Manager server software also requires root privileges for installation.
- **Download non-root `tar` file** containing the binary images for a specific Linux distribution. This is referred to as the **non-root tar file** in this document. Use the non-root `tar` file format ([see section 3.10](#)) in cases where root access to the system is not readily available. This method is easier to use with a nodelocked subscription, as this type of subscription does not require root access.

- **Download ISO image** containing the binary and source images for all Linux distributions. The ISO image contains the RPM, DEB, non-root `tar` installation formats, and the source for files. You can choose to install using either the RPM or non-root tar installation method from this media. This structure is referred to as the **CD image** in this document.
- **A physical CD.** This structure is identical to the contents of the download ISO image. A physical CD is not part of the normal distribution and may be purchased separately.

Table 3-2. Installation Media Contents

ISO/CD Image	Download tar file (RPM/DEB Install)	Download non-root tar file (non-root tar install)
Documentation	Documentation	Documentation
RPM distribution	RPM distribution	-
DEB distribution	DEB distribution	-
Source files	-	-
Subscription server (RPM) tar file distribution	Subscription server (RPM) tar file distribution	Subscription server (RPM) tar file distribution

Additionally, there is an Installer tool which will install the contents of any of the media discussed above. Instruction for using the Installer are in [section 5](#).

3.3

Subscription Management

The Compiler Suite uses subscription management to help users comply with the terms of their subscription agreement. Short descriptions of the types of subscription and instructions for obtaining the the subscription file are given here, but for more details, see the following:

- Details on installing the subscription file are given in [section 3.18](#).
- Details on installing the subscription server are given in [section 3.16](#).
- [See appendix A.7](#) for more information on Subscription Manager client server interaction.
- [See appendix A](#) for longer discussion of the subscription model.
- [See appendix B.2](#) for information on troubleshooting subscription management.

3.3.1

Types of Subscriptions

The subscription management software supports three modes of operation:

- **Nodelocked** single user mode - A single named user on a single machine. For this mode you will need a subscription file residing on the machine running the PathScale software.
- **Nodelocked** multiple user mode - Similar to single user mode. A list of authorized usernames is required.
- **Floating** mode - Multiple users on multiple machines. For this mode you will need a subscription file and a running Subscription Manager server located on some (accessible) machine in your network

All of these types of subscriptions make use of *subscription management client software*, which is installed automatically as a part of the compiler or debugger software.

The subscription management client software requires a *valid subscription file*. This file is not supplied with the software distribution, but is obtained separately from PathScale.

3.3.2

Obtain the Subscription File

To obtain a subscription file for PathScale subscription management, you need to register online to either purchase a subscription or obtain a 30-day trial subscription.

When the requested information has been received, PathScale will send you a file via email that provides the subscription software with the information needed to enable the products. For more information, see this website location:

<http://www.pathscale.com/buy.html>

When you register for your subscription, you will be asked to provide the following information about your system for your subscription file.

For a **nodelocked** subscription:

1. The MAC address of the machine on which the software will run
2. The hostname of the computer on which the software will be run
3. The username of the user, or users, who will run the software

For a **floating** subscription:

1. The MAC address of the computer on which the *subscription server* will run
2. The hostname of the computer on which the *subscription server* will run
3. The number of users for the subscription

3.3.3

Get the Hostname, IP Address, and MAC Address

Gather your hostname, IP address and MAC address. The IP address for your hostname needs to match one of the IP (inet) addresses given in the `ifconfig -a` output.

Use this command to get your hostname:

```
$ hostname
```

This should return a name in a format that follows your local convention:

```
mycomputer.internal.com
```

Next, get the IP address for your computer:

```
$ hostname -i
```

For this example, let's assume the following output:

```
191.138.3.64
```

To get the MAC address for your computer, type:

```
$ /sbin/ifconfig -a
```

You will see something like:

```
eth0  Link encap:Ethernet HWaddr 0a:E6:5B:F1:17:BD
       inet addr: 191.138.3.64 Bcast:192.138.6.255
       Mask:255.255.255.0 UP BROADCAST RUNNING
       MULTICAST MTU:1500 Metric:1
```

```
.  
.  
.
```

The MAC address appears as 6 hex digit pairs, separated by colons, in this case:

```
0a:E6:5B:F1:17:BD
```

In this output also look for the inet (IP) address that matches the one given by the `hostname -i` command:

```
inet addr: 191.138.3.64
```

With floating subscriptions, it is required that the hostname corresponds to the ethernet address in the subscription.

If the machine has multiple active Ethernet interfaces (`eth1`, `eth2`, and so on), the MAC address given to PathScale must match **one** of them. When the compiler is invoked, that interface must be UP (as reported by the `ifconfig` program), or the subscription validation will fail.

3.3.4

Send the Information

After determining the MAC address, IP Address, hostname, and number of users and usernames, send the information via email to support@pathscale.com.

You will be issued a subscription of the type you requested. The subscription file is in a format that is checked by the Subscription Manager server or subscription management client. Most of the fields in the file are subject to a checksum and any modification of these fields will invalidate the subscription. [See section 3.18](#) for more information.

NOTE: The mode (nodelocked or floating) is determined by the PathScale subscription management file (`psscsubscription-<product>*.xml`) that you receive from PathScale. Please contact PathScale if you wish to change your subscription mode.

3.3.5

Installing the Subscription File

Install this file after installing the PathScale Compiler Suite. [See section 3.18](#) for installation instructions.

3.4

Download Software

After determining what method of installation and licensing is appropriate, you can download the software from the PathScale website. Follow the directions on the website for download. A specific download directory is not required.

3.5

Typical Install Scenarios and Terminology

Methods for installing RPMs from the download tar file and the non-root tar file are provided in following sections.

NOTE: Additionally, it is possible to install the software using the PathScale Installer. See [Section 5](#) for more information.

The following table shows the most common installation scenarios. Instructions for each are in the named sections.

Table 3-3. Types of Installation

Installation Type	RPM	DEB	Non-root tar file
Full development environment - contains compilers, runtime, and debugger	3.8	3.9	3.10
Compiler and runtime environment only	4.0.1	4.0.1	-
Debugger and runtime environment only	4.0.2	4.0.2	-
Runtime environment only	4.0.3	4.0.3	-
Network access via remote file system (NFS)	3.12.4	3.12.4	3.12.4

Some terminology is given here.

- `<temporary_directory>` refers to the location into which you have downloaded your software.
- `<distribution_dir>` is the source location from where your software will be installed.
- `<platform_dir>` is specific to your Linux distribution and is a subdirectory of the `<distribution_dir>`.
- `<install_dir>` is be the destination directory into which your software will be installed.

NOTE: The installation instructions are the same for all the Linux distributions.

3.6

Migrating from Previous Releases

The following guidelines assume that you will be installing a new version of the compiler environment to replace an existing version (or beta version) in the same install directory. When you upgrade the compiler environment from a previous release or a beta release, be sure to do the following:

- Before you begin, make a copy of your existing subscription file (`psscsubscription-compiler-*.xml`) and move it to a safe location.
- Remove any beta copies of the release you are going to install to avoid release number conflicts. Then delete the directory tree in which you had the beta release installed.
[See section 6](#) for information on removing the compiler environment.
- Install the new release. [See section 3.8](#) or [section 3.10](#).

- Move your subscription file (`psscsubscription-compiler-*.xml`) to the install directory. See [section 3.19](#).
- Update your `PATH` and `LD_LIBRARY_PATH` to point to executables and shared libraries. See [section 3.12](#).

3.7

Accessing the ISO Image or Physical CD

If you have downloaded the ISO image or are using a physical CD, there is an additional step to perform before installation. You need to select an existing mount point or create a new one. After this is done, you can proceed with either an RPM or non-root tar file installation method as described in [section 3.8](#) and [section 3.10](#).

3.7.1

Mount the ISO Image

To create a new mount point named “iso” use these commands (as root):

```
# mkdir /mnt/iso
(# mkdir -p /mnt/iso will create /mnt if necessary)

# mount -r -o loop <iso_file_name> <mount_point>
```

For example:

```
# mount -r -o loop PathScale-Compiler-Suite-3.2.iso /mnt/iso
```

The `<mount_point>` for your ISO image serves as the `<distribution_dir>` in the rest of this document. Once the image is mounted:

- To install using the binary RPMs, continue to [section 3.8](#).
- To install using the non-root tar file, continue to [section 3.10](#).

3.7.2

Accessing the Physical CD

On many systems mounting a CD will occur automatically when the CD is inserted. If this does not occur, use the following command (as root):

```
# mount /dev/cdrom
```

or

```
# mount -r -t iso9660 /dev/cdrom /mnt/cdrom
```

The point where your CD is mounted (`/mnt/cdrom`) is the same as the `<distribution_dir>` for other media.

Once the image is mounted:

- To install using the binary RPMs, continue to [section 3.8](#).
- To install using the non-root tar file, continue to [section 3.10](#).

NOTE: Some newer distributions have changed the mount point to `/media/cdrom` or `/media/cdrecorder`.

3.8

Installing the Full Development Environment Using RPMs

This section describes the basic RPM installation process. There are multiple interdependent RPM packages that make up the PathScale Compiler Suite. All of the RPMs in the package are necessary to run the compiler. The advantages of using the RPM install method are:

- Automatic installation into default directories
- Uninstall is provided as an option to the `rpm` command
- Environment variables are set automatically via scripts
- Floating licenses can be enabled by installing the Subscription Server RPM

You need to have root privileges to do an RPM install. By default, the compilers are installed into this directory:

```
/opt/pathscale
```

3.8.1

Extract the RPMs from the Download File

After downloading the tarball from the website into `<tmp_directory>`, go there and extract the files from the tarball.

```
$ cd <tmp_directory>
$ tar xvf <download_filename>
```

Extracting the files will create the `<distribution_dir>` which will be similar to this:

```
pathscale-eko-compilers-<version>
```

The `<distribution_dir>` contains documentation, the install tool (see [section 5](#)) and the `<platform_dir>` subdirectory, which is distribution-specific. The `<platform_dir>` contains the directories `rpms` and `subscription-server`. These directories contain the individual RPM files.

NOTE: This extraction step does not need to be done if you are using an ISO Image/Physical CD.

3.8.2

Installing the Compiler RPMs

To install the RPMs for the PathScale Compiler Suite, or upgrade from a previous release, change to the directory for your platform. Become root (as `su -`), and type the following:

```
# cd <distribution_dir>/<your_platform>/rpms
# rpm -Uvh *.rpm
```

NOTE: If you are installing from an ISO image/CD, the <distribution_dir> is the mount point: /mnt/iso or /mnt/cdrom.

The `-Uvh` command performs a new install or upgrades the existing RPMs.

NOTE: If you have trouble installing with the SUSE 9.x distribution, see [section B.1.1](#) for using the “`--nodeps`” argument to work around dependency problems that may occur with some SUSE RPMs.

You will see an installation message similar to the following as the installation progresses (this message is from a SUSE installation).

```
Preparing... ##### [100%]
1:pathscale-compilers-doc ##### [ 8%]
2:pathscale-compilers-lib ##### [17%]
3:pathscale-c ##### [25%]
4:pathscale-compilers-common ##### [33%]
5:pathscale-compilers-devel ##### [42%]
6:pathscale-compilers-lib ##### [58%]
7:pathscale-c++##### [67%]
8:pathscale-f90 ##### [75%]
9:pathscale-sub-client ##### [83%]
10:pathscale-pathdb ##### [92%]
11:pathscale-pathdb ##### [100%]
```

This output will look slightly different during an installation on a Red Hat platform.

3.8.3

Installing for a 32-bit platform

To install the Compiler Suite on a 32-bit platform, use this command:

```
# rpm -Uvh *.i386.rpm
```

NOTE: Due to a deficiency in the version of RPM included in SuSE and SLES distributions, when installing on a 32-bit SuSE or SLES system, the user must specify the `--nodeps` option to `rpm`. This will suppress an erroneous warning that the user does not have 64 bit libraries installed.

3.8.4

Changing the Default Installation Directory

The compiler will be installed under the default <install_directory> which is /opt/pathscale. You can change the default installation directory by typing:

```
# rpm --prefix <install_directory> -Uvh *.rpm
```

<install_directory> will be the name of your new directory.

It should not be necessary to modify either your `PATH` or your `MANPATH` (the scripts in `/etc/profile.d` should set these up). If for some reason, you are having trouble with your `PATH` or `MANPATH`, [see section 3.12](#) for instructions on setting your `PATH` and `MANPATH` to `<install_directory>`.

NOTE: There are some cases where you must uninstall the previous version before installing in a non-default location. Those installing the 3.2 release for the first time are not affected:

- If you previously had a beta distribution installed. After uninstalling, you must then delete the directory tree in which you had the beta release installed.
- If you have performed a relocated installation of the compilers into a non-default directory on supported SUSE and SLES systems
- [See section 6](#) for more information about uninstalling the compilers. Also [see section 3.6](#) for information on migrating from previous releases.

3.8.5

Installing Multiple RPM Versions

You can install and use multiple versions of the compiler simultaneously, either in the same directory, or in different directories.

If you are using RPMs to manage your compiler packages, you *must* use `rpm -i` instead of `rpm -U` to install more than one version of the compiler at a time. If you use `rpm -U`, an attempt to install more than one version will either fail, or will result in all versions except the version you are installing being removed.

You must add the `--force` option to the RPM command line to force it to install more than one version of the compilers at one time.

For example, to install the 3.2 release of the PathScale compilers along with an existing 2.5 release, change to the appropriate directory for your platform and type:

```
$ rpm -ivh --force *.rpm
```

To install in different directories, use the `prefix` option when installing the RPMs.

```
$ rpm --prefix <install_directory> -ivh --force *.rpm
```

See [section 3.8.4](#) for more information.

3.8.6

Options for Multiple RPM Installs

If you have multiple versions installed in the same directory, you can invoke the most recently installed compiler using `pathcc` (or `pathCC`, or `pathf95`). This command will invoke the latest version of the C compiler:

```
<install_directory>/bin/pathcc
```

If you want to invoke a specific version that might not be the most recently installed, use its version number in the command. For example, the first command will run the 2.5 compiler and the second will run the 3.2 compiler :

```
$ <install_directory>/bin/pathcc-2.5  
$ <install_directory>/bin/pathcc-3.2
```

NOTE: With multiple versions installed, the `<install_directory>/bin/pathcc` command will invoke the version of the compiler that you installed most recently (and not necessarily the most recent PathScale release). Use the "`pathcc -v`" command to find out which version is installed as `<install_directory>/bin/pathcc`.

You can also install multiple versions of the compiler in separate directories. You can then set your shell's `PATH` to pick up one version of the compiler by default, and use another version by giving the complete `PATH` to it when compiling software.

See [section 3.12](#) for information on updating your `PATH` and `LD_LIBRARY_PATH` to point to executables and shared libraries.

A note about installing multiple versions with this command:

```
# rpm -i --prefix /<install_directory>
```

The install changes the `/etc/profile.d/pathscale-compilers.sh` and `pathscale-compilers.csh` files, overwriting the current version. You may not notice this behavior if you explicitly set your search `PATH`, ignoring the information in `/etc/profile.d/`.

3.8.7

Files Modified by RPM Install

The following files are created during an RPM installation:

```
/etc/profile.d/pathscale-compilers.sh
```

bash script to set the path

```
/etc/profile.d/pathscale-compilers.csh
```

csh script to set the path

```
/etc/init.d/pathscale-sub
```

Script to start subscription server

The following files are modified:

```
/etc/ld.so.conf.d/pathscale.conf
```

Entry added for PathScale libraries

The installation does not modify `LD_LIBRARY_PATH` or `MANPATH`.

3.8.8

Verify Subscription File

After finishing the installation, verify that your subscription file is correctly configured (either in nodelocked mode or in conjunction with the Subscription Manager) before running the compiler. The compiler will not run without a valid and correctly installed subscription file. Proceed to [section 3.13](#) for more information.

3.9

Installing the Full Development Environment Using DEBs

This section describes the basic DEB installation process. There are multiple interdependent DEB packages that make up the PathScale Compiler Suite. All of the DEBs in the package are necessary to run the compiler. The advantages of using the DEB install method are:

- Automatic installation into default directories
- Uninstall is provided as an option to the `dpkg` command
- Floating licenses can be enabled by installing the Subscription Server DEB

You need to have root privileges to do an `dpkg` install. By default, the compilers are installed into this directory:

```
/opt/pathscale
```

NOTE: The compiler suite are 32-bit executables and require a 32-bit execution environment. In some cases, the required 32-bit packages are not installed. Minimally, the following need to be installed to run the compilers: `lib32gcc`, `lib32stdc++`, `libc6-i386`.

3.9.1

Extract the DEB from the Download File

After downloading the tarball from the website into `<tmp_directory>`, go there and extract the files from the tarball.

```
$ cd <tmp_directory>
$ tar xvf <download_filename>
```

Extracting the files will create the `<distribution_dir>` which will be similar to this:

```
pathscale-eko-compilers-<version>
```

The `<distribution_dir>` contains documentation, the install tool (see [section 5](#)) and the `<platform_dir>` subdirectory, which is distribution-specific. The `<platform_dir>` contains the directories `rpms` and `subscription-server`. These directories contain the individual DEB files.

NOTE: This extraction step does not need to be done if you are using an ISO Image/Physical CD.

3.9.2

Installing the Compiler DEB

To install the DEBs for the PathScale Compiler Suite, or upgrade from a previous release, change to the directory for your platform. Become root (as `su -`), and type the following:

```
# cd <distribution_dir>/<your_platform>/debs  
# dpkg -i *.deb
```

NOTE: If you are installing from an ISO image/CD, the `<distribution_dir>` is the mount point: `/mnt/iso` or `/mnt/cdrom`.

You will see an installation message similar to the following as the installation progresses.

```
Selecting previously deselected package pathscale-c.  
(Reading database ... 118188 files and directories currently  
installed.)  
Unpacking pathscale-c (from pathscale-  
c_3.2.915.15307.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-c++.  
Unpacking pathscale-c++ (from pathscale-  
c++_3.2.915.15307.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-compilers-  
common.  
Unpacking pathscale-compilers-common (from pathscale-compilers-  
common_3.2.915.15307.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-compilers-devel.  
Unpacking pathscale-compilers-devel (from pathscale-compilers-  
devel_3.2.915.15307.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-compilers-docs.  
Unpacking pathscale-compilers-docs (from pathscale-compilers-  
docs_3.2.915.15307.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-compilers-libs.  
Unpacking pathscale-compilers-libs (from pathscale-compilers-  
libs_3.2.915.15307.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-compilers-libs-  
lgpl.  
Unpacking pathscale-compilers-libs-lgpl (from pathscale-compilers-  
libs-lgpl_3.2.915.15307.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-f90.  
Unpacking pathscale-f90 (from pathscale-  
f90_3.2.915.15307.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-pathopt2.  
Unpacking pathscale-pathopt2 (from pathscale-  
pathopt2_3.2.506.97.ubuntu7.10.psc_amd64.deb) ...  
Selecting previously deselected package pathscale-sub-client.  
Unpacking pathscale-sub-client (from pathscale-sub-  
client_3.2.844.469.ubuntu7.10.psc_amd64.deb) ...  
Setting up pathscale-c (3.2.915.15307.ubuntu7.10.psc) ...  
Setting up pathscale-c++ (3.2.915.15307.ubuntu7.10.psc) ...  
Setting up pathscale-compilers-common  
(3.2.915.15307.ubuntu7.10.psc) ...  
Setting up pathscale-compilers-devel  
(3.2.915.15307.ubuntu7.10.psc) ...  
Setting up pathscale-compilers-docs (3.2.915.15307.ubuntu7.10.psc)  
...  
Setting up pathscale-compilers-libs (3.2.915.15307.ubuntu7.10.psc)  
...  
  
Setting up pathscale-compilers-libs-lgpl  
(3.2.915.15307.ubuntu7.10.psc) ...  
Setting up pathscale-f90 (3.2.915.15307.ubuntu7.10.psc) ...  
Setting up pathscale-pathopt2 (3.2.506.97.ubuntu7.10.psc) ...
```

```
Setting up pathscale-sub-client (3.2.844.469.ubuntu7.10.psc) ...  
Processing triggers for libc6 ...  
ldconfig deferred processing now taking place
```

3.9.3

Files Modified by DEB Install

The following files are created during an RPM installation:

```
/etc/profile.d/pathscale-compilers.sh
```

bash script to set the path

```
/etc/profile.d/pathscale-compilers.csh
```

csh script to set the path

```
/etc/init.d/pathscale-sub
```

Script to start subscription server

The following files are modified:

```
/etc/ld.so.conf.d/pathscale.conf
```

Entry added for PathScale libraries

The installation does not modify LD_LIBRARY_PATH or MANPATH.

3.9.4

Verify Subscription File

After finishing the installation, verify that your subscription file is correctly configured (either in nodelocked mode or in conjunction with the Subscription Manager) before running the compiler. The compiler will not run without a valid and correctly installed subscription file. Proceed to [section 3.13](#) for more information.

3.10

Installing the Full Development Environment as a Non-root User

This section describes installing a tar file. The advantage of using this method is that you do not have to have root privileges to install the PathScale Compiler Suite.

NOTE: If you are installing from a tar file and using a floating subscription, you will need root privileges to install the floating license server software for the Subscription Manager. This software is available as an RPM install only. [See section 3.16](#).

The installation process for a non-root user is the same for all supported Linux distributions.

3.10.1

Extract the Non-root tar File

Note that this type of installation will install into the current working directory rather than a default location. For this example we assume that you have created your own directory called `<my_compiler_directory>`.

After downloading the tarball from the website into `<temporary_directory>`, go to the directory and extract the files from the tarball.

```
$ cd <temporary_directory>
$ tar xvf <download_filename>
```

NOTE: This extraction step does not need to be done if you are using an ISO Image/Physical CD.

Extracting the files will create the `<distribution_dir>` which will be similar to this:

```
pathscale-eko-compilers-<version>
```

This directory contains documentation, the install tool and the distribution-specific `<platform_dir>` subdirectory. The `<platform_dir>` contains the directories `tar` and `subscription-server`. The `subscription-server` directory contains the subscription server in RPM format. The `tar` directory contains the individual `tar` file that has been compressed using the `bzip2` compression program. It will look similar to this:

```
pathscale-compilers-redhat3-3.2-74.2093.tar.bz2
```

This follows this convention:

```
pathscale-compilers-<platform>-<version>.tar.bz2
```

The `bzip2` compression program is installed as standard on all Linux machines. If for some reason `/usr/bin/bzip2` does not exist on your system, it is available on the distribution media that came with your operating system.

3.10.2

Installing the Compilers from a Non-root tar File

To install the non-root tar file for the PathScale Compiler Suite, or upgrade from a previous release, change to the directory where you want to install the software. In this example it is called `my_compiler_directory`.

For example:

```
$ cd /my_compiler_directory
$ tar xjf <path_to_tarfile>/
  pathscale-compilers-redhat3-3.2-74.2093.tar.bz2
```

Note that for an ISO image the first part of the `<path_to_tarfile>` will be `/mnt/iso`.

The `tar` file will be unpacked into a directory whose name looks like this:

```
pathscale-compilers-redhat3-3.2-74.2093
```

You can rename this directory or copy it to another directory, but you must not rearrange the structure of anything under it, or the compiler may not work.

Inside this directory, the structure is as follows:

- `bin` - the compiler binaries
- `man` - the man pages
- `share/doc` - other docs

All the other subdirectories contain supporting programs, headers, and libraries.

NOTE: This does not install the Subscription Manager server, which is required if a floating subscription is being used. The Subscription Manager server software can only be installed using the RPM method. [See section 3.13](#) for more information on installing the Subscription Manager software.

3.10.3

Installing Multiple Non-root tar File Versions

If you are using a non-root `tar` file to install your compilers, you can simply extract your `tar` file and copy the files to wherever you need. Since `tar` cannot automatically remove files for you, we recommend that if you are using `tar`, you install different versions of the compilers into different directories, instead of all into one directory.

3.11

Installing Subsets of the Compiler Suite

You may not want to install the entire PathScale Compiler Suite environment. [See section 4](#) for information on installing subsets of the compiler environment; the individual compilers, the debugger, or just the runtime environment.

3.11.1

After Installation

To run the compilers, set up your shell's `PATH` so it contains the `PATH` to wherever you installed the compilers (`<install_directory>`), and make sure the Subscription File is installed.

Proceed to [section 3.12](#) for more information on setting your `PATH` and other environment variables. [Section 3.18](#) discusses setting up the Subscription File.

3.12

Setting Environment Variables

- **With an RPM install**, a file called `/etc/ld.so.conf.d/pathscale.conf` is automatically updated to point at the locations of the shared versions of the runtime libraries. The `ldconfig` program is run to tell the dynamic linker about these new directories. Your `PATH` and `MANPATH`, should have been automatically set as well.
- **With a non-root tar file install**, you need to configure the dynamic linker by hand, and set your `PATH` and `MANPATH`.

Instructions for setting these and other variables are given in the following sections.

3.12.1

Using Shared Libraries

If you are installing a non-root `tar` file, you must configure the dynamic linker by hand. There are two ways to do this, depending on whether you have root privileges.

1. If you do not have root privileges, use this method with a `bash` shell:

```
$ export \  
LD_LIBRARY_PATH=<install_directory>/lib/<version>: \  
<install_directory>/lib/<version>/32:$LD_LIBRARY_PATH
```

Using a `csh` shell, the command would look like this:

```
$ setenv \  
LD_LIBRARY_PATH <install_directory>/lib/<version>: \  
<install_directory>/lib/<version>/32:$LD_LIBRARY_PATH
```

If you are going to run compiled code on a different machine, see [section 4.0.3](#) for information on runtime installation.

2. If you have root privileges, use this method.

Create a file in `/etc/ld.so.conf.d` named `pathscale-compilers.conf` that contains the following lines (the order in which they appear does not matter):

```
<install_directory>/lib/<release>  
<install_directory>/lib/<release>/32
```

After adding these lines, run the `/sbin/ldconfig` program to tell the dynamic linker about these new directories. You will need root privileges to do this.

```
# /sbin/ldconfig
```

If you uninstall a non-root `tar` file release of the compilers, remove the file `/etc/ld.so.conf.d/pathscale-compilers.conf` and rerun `ldconfig`.

For more information on `ldconfig`, see the `ldconfig` man page.

3.12.2

Setting the `PATH`

If you installed with RPMs, your `PATH` should have been automatically set. If you installed from a non-root `tar` file, you must set your path so that it contains the path to wherever you installed the compilers. It will be in this format:

```
<install_directory>/bin
```

You can manually set the `PATH` for the compilers with the following (using a `bash` shell). This example uses the default `<install_directory>`:

```
$ export PATH=$PATH:/opt/pathscale/bin
```

To set the `PATH` using the `cs` shell, type:

```
$ setenv PATH $PATH:/opt/pathscale/bin
```

3.12.3

Access to Compiler man Pages with `MANPATH`

Depending on the distribution of Linux the user may or may not have access to the PathScale Compiler Suite man pages by default. On all Linux distributions, end users can set their `MANPATH` environment variable to include `<install_directory>/man` to get access to the man pages.

These examples use the default `<install_directory>`

To set the `PATH` using the `bash` shell, type:

```
$ export MANPATH=${MANPATH}:/opt/pathscale/man
```

To set the `PATH` using the `cs` shell, type:

```
$ setenv MANPATH $MANPATH:/opt/pathscale/man
```

An alternative is to update the system `config` files for this path. The method for RedHat/Fedora is slightly different than for SUSE.

On Red Hat and Fedora distributions, edit the following file (as root):

```
/etc/man.config
```

If `NOAUTOPATH` has been uncommented, then the following line should be added to the file:

```
MANPATH /opt/pathscale/man
```

You should change `/opt/pathscale` to the actual install location.

If `NOAUTOPATH` is commented out, and `<install_dir>/bin` for the compiler environment is in your `PATH`, then the `man` command should work.

On SUSE distributions, edit this file (as root):

```
/etc/manpath.config
```

Add this line:

```
MANDATORY_MANPATH /opt/pathscale/man
```

You should change `/opt/pathscale` to the actual install location.

Once these changes are done, either log out and log back in or run the `manpath` command to obtain the new setting.

3.12.4

Setting Up the Compilers for Use Over NFS

If you are installing the PathScale compilers to be used over a network using a distributed file system, you will need to set your environment variables for subscription management to point to the compiler and subscription file. See [section 3.16](#) for more information about setting those environment variables.

If you install the compiler environment using RPMs, a file called `/etc/ld.so.conf` is automatically updated to point at the locations of the shared versions of the runtime libraries on the machine you installed on. You need these libraries to run the compiler. The `ldconfig` program is run to tell the dynamic linker about these new directories.

If you are installing for use over a distributed network or installing from a tar file, you must update the locations of the shared versions of the runtime libraries on the machine you installed on by hand. You will also need to set the `PATH` (see [section 3.12.2](#)) and `MANPATH` (see [section 3.12.3](#)).

NOTE: The procedure for updating the directory locations is similar to the one you would use for a non-root `tar` file installation in [section 3.12.1](#).

There are two ways to set the library paths, depending on whether or not you have root privileges.

1. If you do not have root privileges, use this method with a `bash` shell:

```
$ export \  
LD_LIBRARY_PATH=<install_directory>/lib/<version>: \  
<install_directory>/lib/<version>/32:$LD_LIBRARY_PATH
```

Using a `cs` shell, the command would look like this:

```
$ setenv \  
LD_LIBRARY_PATH <install_directory>/lib/<version>: \  
<install_directory>/lib/<version>/32:$LD_LIBRARY_PATH
```

If you are going to run compiled code on a different machine, see [section 4.0.3](#) for information on runtime installation.

2. If you have root privileges, use this method.

Add the following lines to `/etc/ld.so.conf` (the order in which they appear does not matter):

```
<install_directory>/lib/<release>  
<install_directory>/lib/<release>/32
```

After adding these lines, run the `/sbin/ldconfig` program to tell the dynamic linker about these new directories. You will need root privileges to do this.

```
# /sbin/ldconfig
```

If you uninstall the compiler (for a client installation), delete these lines from `/etc/ld.so.conf` and rerun `ldconfig`.

For more information on `ldconfig`, see the `ldconfig` man page.

3.12.5

Setting and Using NLSPATH

Many compilers use the `NLSPATH` environment variable, the full path name for message catalogs. The default is usually something like:

```
NLSPATH=/usr/share/nls/%N.cat
```

If this variable is set with the PathScale compilers, the compiler will simply exit with a message like:

```
pathf95 INTERNAL: Unable to open message system.
```

To fix this, either un-set the `NLSPATH` environment variable, or set the `NLSPATH` environment variable to the location of the error messages, using `%N` for the base name of the file. Modify your `NLSPATH` to read something like this:

```
<old_nls_path>:<eko_install_directory >/lib/2.1.99/%N.cat
```

`old_nls_path` is the previous value of this variable, and `eko_install_directory` is the directory where you installed the PathScale compilers (usually `/opt/pathscale`).

3.13

PathScale Subscription Management

PathScale subscription management consists of three primary components:

- Subscription Manager Client for both nodelocked and floating modes
- Subscription Manager Server for floating mode
- Subscription File for both nodelocked and floating modes

If you are running in nodelocked mode, you can skip to [section 3.18](#) to install the Subscription File.

3.14

Installing the Subscription Manager Client

The subscription management client software is installed along with the compiler or debugger RPMs or non-root `tar` file. **If you have already installed the compiler software, the subscription management *client* (also referred to as the subclient) has already been installed for you.**

If you are using an RPM install, you can check to see if the subclient has been installed, and if not, selectively install the single RPM. To check if it has been installed type:

```
$ rpm -qa | grep sub-client
```

The RPM will look something like:

```
pathscale-sub-client-2.3-25.116.suse9.0_psc
```

You can also check in `<install_directory>/lib/subclient/<version>` to see if the subclient is there. If, for some reason, the RPM did *not* get installed, you will need to find the `pathscale-sub-client*` RPM and install it.

To install the subscription management subclient, go to your PathScale software distribution, and change to the directory containing the subscription management software for your platform. You must become root first (we recommend `su -`).

```
# cd <distribution>/<platform>/pathscale-sub-client*
```

```
# rpm -Uvh pathscale-sub-client*.rpm
```

This will install the subclient in the default prefix location:

```
/opt/pathscale/lib/<release>
```

NOTE: The subscription management version number is unrelated to the software version (or release) number.

3.15

About Floating Subscriptions

A floating subscription can be tied to a named set of users. The users can be either exclusive or non-exclusive. The modes work like this:

- **Exclusive users:** In this mode, only the named users will be granted leases for use. No other users will be allowed to use the software.
- **Non-exclusive users:** Like the exclusive mode, a list of users can be specified. However, other users will be allowed to use the software, but with a lower priority than those users who are known to the Subscription Manager server. If a named (priority) user wishes to gain a lease, a non-priority user may lose his or her lease in preference to the named user. This will only happen if there are no available leases. The unnamed user chosen to lose the lease will be the one who has the least time to run on the lease.

The non-exclusive user configuration is useful for automated builds, where you want to guarantee the build will be granted a lease. The priority user scheme allows a core set of users to be specified to be given the highest priority, but also allows others to use the PathScale software when there are available leases not being used by the core users.

The set of named users, and the exclusivity of the set is contained in the subscription file, and can be modified by the administrator to add, or remove users, and change the exclusivity. [See section 3.18](#) for more information on specifying these modes.

3.16

Installing the Subscription Manager Server for Floating Mode

To run the PathScale subscription management in floating subscription mode, you will need to install and configure the Subscription Manager *server* software. This software will need to be installed on the machine that will be the server. It is only available as an RPM install.

NOTE: The choice of mode is determined by the PathScale subscription management file (`psscsubscription-<product>-* .xml`) that you received from PathScale. Please contact PathScale if you wish to change your subscription mode.

To install the Subscription Manager server software, you need to have root privileges. In your PathScale software distribution, change to the directory containing the Subscription Manager server software for your platform. You must become root first (we recommend `su -`).

```
# cd <distribution>/<platform>/subscription-server
# rpm -Uvh pathscale-sub-server*.rpm
```

This will install the program in the default prefix location (`/opt/pathscale/lib/<release>`). To install in a different location, the `--prefix` option may be used.

Once the software is installed, you need to install the subscription file itself (`psscsubscription-<product>-* .xml`). [See section 3.18](#) for information on installing for floating mode.

3.17

Configuring the Environment for Floating Subscriptions

This section provides information on environment variables and the subscription daemon.

3.17.1

Environment Variables for Subscription Management

The following environment variables may be used with the PathScale Subscription Manager server.

- **PATHSCALE_SUBSCRIPTION_DAEMON** The location where the daemon is running. This is of the format `addr[:port]`. The address is either a hostname or an IP address (`n.n.n.n`). The port is optional and defaults to 7143.
- **PATHSCALE_SUBSCRIPTION_DIR** The location of `pscsubscription-<product>-.xml` file. This specifies the directory containing the file.
- **HTTP_PROXY_HOST** The hostname of the proxy machine.
- **HTTP_PROXY_PORT** The port on which the proxy is running.

3.17.2

The Subscription Daemon

The subscription daemon is a process running in the background that checks the status of the subscription. The floating subscription model uses the concept of a 'seat'. This is defined to be a unique username and hostname pair. When you buy a floating subscription, PathScale needs to know how many seats you will require. The subscription daemon keeps track of the number of seats you have used at any time and how many are available.

The subscription daemon uses the following formula to determine if there are seats available:

$$\text{numseats} = \max(\text{users}, \text{machines})$$

For example, if a site has 10 users and 20 machines, and the subscription specifies that the number of seats is 10, then 10 users can execute on 10 or less machines.

Like the Nodelocked subscription, the floating subscription daemon is tied to a single host, specified by the MAC address. This means that the daemon cannot be run on another host. If this is required, another subscription file must be generated.

When a user runs the compiler that has a floating subscription, the compiler must be able to find the daemon. The following algorithm is used:

1. First see if it can find the subscription file. It searches for the a file called `pscsubscription.xml` in the following places:
 - a. The contents of the environment variable `PATHSCALE_SUBSCRIPTION_DIR`
 - b. The user's home directory
 - c. The location in which the compiler was installed
2. If the subscription file was found, look inside it to see if the `<daemon/>` tag was specified. If so, use this to determine the host and port of the server
3. If the subscription file was not found, look for the server on the following addresses:
 - a. The contents of the environment variable `PATHSCALE_SUBSCRIPTION_DAEMON`
 - b. The hostname 'pscsubscriptionserver' (if it exists)

The floating subscription daemon runs by default on port TCP/7143, but this can be changed when the subscription is issued. It is not possible to change the port number after the subscription file is generated without generating a new file.

If the port number is changed from the default, the compiler must be told what the new port number is. This is done either by allowing the compiler to see the subscription file or by specifying the hostname and port in the `PATHSCALE_SUBSCRIPTION_DAEMON` environment variable as outlined by the rules above.

3.17.3

Proxies

If the subscription daemon is running on a machine that is on the other side of a firewall from the machines running the PathScale software or tools, it may be necessary to use an HTTP proxy to talk to it. This can be achieved by setting the following environment variables in the shell of the user running the software:

- `HTTP_PROXY_HOST` - The hostname of the proxy machine
- `HTTP_PROXY_PORT` - The port on which the proxy is running

The software will attempt to connect to the daemon directly first before using the proxy.

3.17.4

Running the Daemon on a Different Interface

Some users prefer that the daemon run and listen on a single Ethernet interface, rather than all of the available interfaces. By default, the subscription daemon listens on all the interfaces available to it. If this is not the desired behavior, you can change the address the daemon listens on by passing the flag `-iface n.n.n.n` to the subserver program when it starts. The value of `n.n.n.n` is the IP address assigned to the interface you want to listen on.

To specify an IP address using `-iface` you will need to edit the file `/etc/rc.d/init.d/pathscale-sub`. If the IP address you want to use is `130.126.230.190`, you would add the `-iface 130.126.230.190` argument to this line:

```
daemon --user=psc_sub $prefix/subserver -d
```

So that it becomes:

```
daemon --user=psc_sub $prefix/subserver -d -iface 130.126.230.190
```

Now the subscription daemon will only listen on port `130.126.230.190`.

NOTE: It is important that the MAC address given to PathScale when the subscription is issued be an active interface.

3.17.5

Running the Subscription Server

Use these commands to control the Subscription Server.

You can set the Subscription Manager server so it will start every time a server boots with the following command (as root):

```
# /sbin/chkconfig pathscale-sub on
```

To turn subscription server off so it doesn't start when the server boots, type (as root):

```
# /sbin/chkconfig pathscale-sub off
```

Use the following commands to manually start, stop, restart, and get status. For RedHat/Fedora:

```
$ /sbin/service pathscale-sub [ start|stop|restart|status ]
```

For SuSE and SLES, the `/sbin/service` command does not exist, so type:

```
$ /etc/init.d/pathscale-sub [ start|stop|restart|status ]
```

3.18

The Subscription File

The PathScale Compiler Suite will not run without a valid and correctly installed subscription file, whether you are using a floating subscription or a nodelocked subscription. You will need to install this subscription file before you can start using the PathScale software. This file is named:

```
pscsubscription-<product>-* .xml
```

This file is not supplied with the PathScale software, but is obtained separately from PathScale. [See section 3.3.2](#) for information on obtaining your subscription file.

The file is in an XML-like format and can be edited by the administrator. Beware that most of the fields in the file are subject to a checksum and any modification to these fields will invalidate the subscription.

Where you install the subscription file depends on the type of subscription you have purchased from PathScale; floating or nodelocked.

For floating subscriptions, the following field can be edited by the administrator:

```
<users value="" exclusive="" />
```

This is the list of users and the exclusivity setting. The list of users is a comma-separated string of usernames (no spaces allowed). The value of `exclusive` is either `true` or `false`.

For example:

```
<users value="dave,joeb,alice1,bob" exclusive="false"/>
```

For nodelocked subscriptions, the “`users value`” field cannot be changed by the administrator. An updated subscription file must be requested from PathScale.

Depending on the subscription model, the subscription file must be available to certain components of the subscription software:

- Nodelocked: The file must be readable by the user running the PathScale software.
- Floating: The file must be readable by the subscription server (daemon). If the client will be reading the file to obtain the server name and port, then the file must be readable by the user running the PathScale software.

NOTE: Failing to correctly install the subscription file will cause your PathScale software tools or compiler to exit with an error message.

For more detailed background information on the floating subscription or nodelocked subscription models, [see appendix A](#).

3.18.1

Subscription File for Nodelocked Subscriptions

To correctly install the subscription file, you will need to do *one* of the following. Once the file is installed, you can start using the PathScale software.

1. Copy the file into the same directory as the subscription software. For example, to install your `pscsubscription-<product>-* .xml` file (from the email you received from PathScale) into the same directory as subscription management client you installed with the EKOPath Compiler Suite (the `/opt/pathscale/lib/3.2` directory), you would type:

```
$ cp pscsubscription-<product>-* .xml /opt/pathscale/lib/3.2/
```

You may need root privileges to install the file in some locations.

2. Put the file into `~/ .pathscale`

```
$ cp pscsubscription-<product>-* .xml /home/.pathscale
```
3. Set the environment variable `PATHSCALE_SUBSCRIPTION_DIR` *<path-to-directory-containing-file>* to point to the directory containing the `pscsubscription-<product>-* .xml` file.

3.18.2

Subscription File for Floating Subscriptions

To install for floating subscription mode:

1. To set up priority or exclusive users when using a floating subscription, update the `pscsubscription-<product>-* .xml` file as described in [section 3.18](#).
2. Start the Subscription Manager

```
$ /etc/init.d/pathscale-sub start
```

3. On the client side, one of the following must be true and configured for the user:
 - a. Set the environment variable `PATHSCALE_SUBSCRIPTION_DAEMON` pointing to the server and port the Subscription Manager server is running on, for example:

```
$ setenv PATHSCALE_SUBSCRIPTION_DAEMON
155.101.16.88:7143
```
 - b. The `pscsubscription-<product>-*.xml` file with the correct server name and port number must be located in:

```
<install_directory>/pathscale/lib/<version>/
```

NOTE: The server name and port number must match the `<daemon/>` tag in the subscription file, if one has been set.

The environment variable `PATHSCALE_SUBSCRIPTION_DIR` may be set prior to invoking the server to point to the directory containing the subscription file if it is in a different location. See [section 3.12](#) for more information on environment variables.

NOTE: If you are installing multiple versions of the PathScale EKOPath compilers, the subscription file must be installed in the same directory as the subscription software or you must set the environment variable `PATHSCALE_SUBSCRIPTION_DIR` to point to the directory containing the file.

3.19

Upgrading an Existing Installation

If you are installing a new release of the PathScale EKOPath Compiler Suite or other PathScale software, and you have an existing subscription that will work with the new release, you need to change the location of your subscription file to work with the new release.

Copy the `pscsubscription-<product>-*.xml` file from:

```
<install_dir>/lib/<old_version>/pscsubscription-<product>-*.xml
```

To this new location:

```
<install_dir>/lib/<new_version>/pscsubscription-<product>-*.xml
```

For example, if you had a 2.5 release of the compiler, and you were upgrading to the 3.2 release, you would copy this file:

```
<install_dir>/lib/2.5/pscsubscription-<product>-*.xml
```

To this new location:

```
<install_dir>/lib/3.2/pscsubscription-<product>-*.xml
```

If you have your subscription file in `~/ .pathscale` or have set the `PATHSCALE_SUBSCRIPTION_DIR` environment variable to point to the directory containing the subscription file, you should not have any difficulty with an upgrade. See sections [3.18](#), [3.18.1](#), and [3.18.2](#), for more details on correctly installing the subscription file.

3.20

Monitoring Lease Usage

If you are using the subscription server, you can access current lease information about your PathScale subscription from a web browser. To see the current lease usage, paste this URL into your web browser:

```
http://pscsubscriptionserver:7143/
```

The URL is made up of the hostname for your subscription server and the port number on which the server is running. If you haven't created `pscsubscriptionserver` in your DNS (see [section 3.17.2](#)), you will have to substitute the actual hostname that your subscription server is running on. Port 7143 is the default port number; if you have changed your port number, use that number instead. This will not work if you are using a nodelocked subscription.

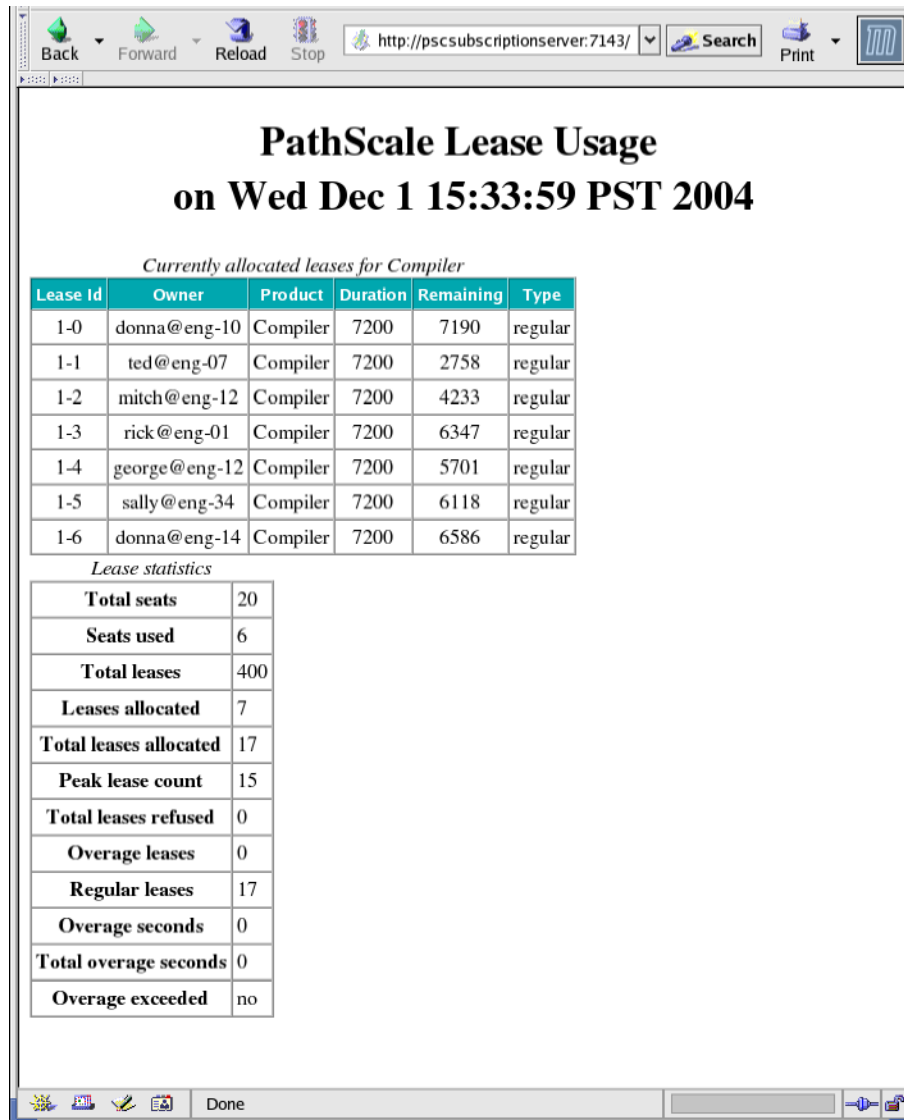


Figure 3-1. Example of Lease Usage Page

3.21

Verifying and Running the Compiler

Log out and log back in to your machine, and then check your PATH.

```
$ pathcc -v
PathScale(TM) Compiler Suite: Version 3.2
Built on: 2007-10-11 06:32:23 -0800
Thread model: posix
GNU gcc version 4.2.0 (PathScale 3.2 driver)
```

If the `pathcc -v` command produces output similar to that above, everything should be working correctly.

If your `PATH` is not set correctly see the directions in [section 3.12.2](#) for setting and checking your `PATH`.

Once the `PATH` has been correctly set, the compilers can be invoked using the following names:

- `pathcc` The PathScale C compiler
- `pathCC` The PathScale C++ compiler
- `pathf95` The PathScale Fortran compiler

3.21.1

Hello World

Try compiling Hello World or some other very simple program, and then your own code. Here's a test using `hello.c`.

```
$ cat hello.c
#include <stdio.h>
int main () {
    int i;
    printf("Hello from PathScale\n");
    return 0;
}

$ export PATH=/opt/pathscale/bin:${PATH}
$ pathcc -o hello hello.c
$ ./hello
Hello from PathScale
```

See the *PathScale Compiler Suite User Guide* for information on using the compiler and optimizing your code.

Section 4

Installing Subsets of the PathScale Compiler Suite

You may not want to install the entire PathScale Compiler Suite environment. This section covers installing subsets of the compiler environment; the individual compilers, the debugger, or just the runtime environment.

Refer to [section 3](#) for instructions about environment variables, subscription management and so on. You will still need a valid subscription to run subsets of the Compiler Suite.

4.0.1

Installing Individual Compilers and Runtime Environment

The PathScale Compiler Suite has a number of supporting packages that must be installed for the compilers. This list includes the RPMs named:

- `pathscale-compilers-common-*`
- `pathscale-compilers-docs-*`
- `pathscale-compilers-libs-*`
- `pathscale-sub-client-*`

The remaining packages consist of the following:

Component	Package
C compiler	<code>pathscale-c-*</code>
C++ compiler	<code>pathscale-c++-*</code>
Fortran compiler	<code>pathscale-f90-*</code>
pathdb	<code>pathscale-pathdb-*</code>
pathopt2	<code>pathscale-pathopt2-*</code>
Subscription Server Manager Server	<code>pathscale-sub-server-*</code>

After you have installed the other packages, then install the appropriate package for the language you want. For example, to install the Fortran compiler you would type:

```
# rpm -Uvh pathscale-f90-*
```

or, on Ubuntu:

```
# dpkg -i pathscale-f90-*
```

NOTE: Installing the individual compilers only works with the package install.

4.0.2

Installing the PathScale Debugger and Runtime Environment

If you used the full package installation (see section 3.8), you already installed the PathScale debugger, `pathdb`, since it is part of your compiler environment.

If you used the non-root tar file install method, `pathdb` was installed automatically. You will find the PathScale debugger in the `<install_directory>/bin` directory.

4.0.2.1

Installing PathDB Only

If you need to install the debugger alone, you will find the PathScale debugger package for your platform in the directory for that platform.

```
<distribution_dir>/<platform>/<pkg-type>
```

For example, for the Fedora Core 3 platform, you would find the `pathdb` RPMs here:

```
<distribution_dir>/Fedora/rpms
```

NOTE: There is an RPM for the 32-bit debugger (`i386`) and one for the 64-bit debugger (`x86_64`). Choose the appropriate one for your debugging environment. You can also install both RPMs.

You must have root privileges to install the PathScale debugger packages. Obtain root privileges and install the `pathdb` package (for the 32-bit version) using this command:

```
# rpm -Uvh pathscale-pathdb-*i386.rpm
```

or, on Ubuntu:

```
# dpkg -i pathscale-pathdb-i386*.deb
```

For the 64-bit version use this command:

```
# rpm -Uvh pathscale-pathdb-*x86_64.rpm
```

or, on Ubuntu:

```
# dpkg -i pathscale-pathdb-x8664*.deb
```

You will see output similar to the following:

```
# rpm -Uvh pathscale-pathdb-*x86_64.rpm
Preparing... ##### [100%]
1:pathscale-pathdb ##### [100%]
```

To install both the 32-bit and the 64-bit version, use this command:

```
# rpm -Uvh pathscale-pathdb*.rpm
```

NOTE: The PathScale debugger, `pathdb`, requires a compiler subscription to run. In the case of a nodelocked subscription, `pathdb` can only be used on the system specified in the nodelocked license.

4.0.3

Installing the Runtime Environment Only

To run code generated with the PathScale Compiler suite on a machine without a *complete* installation of the compilers, you also need to install this library on your host machine:

- `pathscale-compilers-libs-3.2*.i386.<pkg-type>`

These libraries are shipped as psackages with the PathScale Compiler Suite and can be found with each platform distribution. Install the version of these libraries for your platform.

Section 5

PathScale Installer

With the PathScale Installer, the PathScale Compiler Suite can be installed from RPMs, tar files, or an ISO image to those directories for which you have access and permissions. The Installer can also configure your subscription management and make any necessary library path changes. It can be used to install, upgrade, or install alongside existing installations. The Installer can also be used to remove currently installed compiler environments.

NOTE: The PathScale Installer only works for installing the supported distributions for the PathScale compilers.

- Red Hat Fedora Core 3, 4, 5, 6
- Fedora 7, 8
- Red Hat Enterprise Linux (RHEL) 4 and 5
- SUSE Linux Enterprise 10
- SUSE Linux Enterprise Server (SLES) 9
- SUSE Professional 9.3, 10, 10.1
- openSUSE 10.3

NOTE: The installer does not support Ubuntu at the time of this release. Please check the support website for an updated installer:

<http://www.pathscale.com/support.html>

5.1

Where to Find the Installer

The installer can be found:

- As part of the download distribution (in the RPMs or tar file)
- On the distribution CD (ISO image)
- It can be downloaded separately from the PathScale web site.

5.2

What You Need to Run the Installer

Pre-requisites for using the Installer include:

- A version of the installer
- Access and permissions for any directory containing this software
- Access and permissions for any directory in which you are installing

- Software in the form of RPMs, a `tar` file, or an ISO image (mounted) that you want to install
- A valid Subscription File

NOTE: SUSE 9.x and SLES 9 users must have `python` and the `python-curses` package installed to use the Installer. These packages can be found on your distribution media, but are not part of the default installation. You may want to use the RPM install method instead.

5.3

Downloading Compiler Suite Software

Download the PathScale Compiler Suite software to install from the PathScale support web site.

<http://www.pathscale.com/support/>

5.3.1

Unpack tar File (RPM and Non-root tar Install Only)

After you have downloaded the software, change into the directory where the software is located to unpack the tar file.

If you downloaded a root/RPM tarball, use these commands (as root):

```
# cd <temporary_directory>
# tar xvf <downloadfilename.tar>
```

If you downloaded a non-root install tarball, the commands are the same but you don't need to be root:

```
$ cd <temporary_directory>
$ tar xvf <downloadfilename.tar>
```

In both cases, a directory named `pathscale-eko-compilers-<version>` is created. This is the directory in which you will run the installer.

NOTE: If you downloaded the ISO image or are using a CD, you do not need to use these commands, as these files are not in the tarball format.

5.4

Mounting an ISO Image

If you downloaded the ISO image, you need to have root privileges. You then need to mount the ISO image so that it is accessible. To do this, you need to select an existing mount point or create a new one.

To create a new mount point named `iso` use this command (as root):

```
# mkdir /mnt/iso
# mount -r -o loop <iso_file_name> <mount_point>
```

For example:

```
# mount -r -o loop PathScale-Compiler-Suite-3.1.iso /mnt/iso
```

Once you have mounted the ISO image, you can change directories into the mount point and run the installer.

5.5

Accessing the Physical CD

If you are installing software from a CD, you need access to the CD. On many systems mounting a CD will occur automatically when the CD is inserted. If this does not occur, use the following command (as root):

```
# mount /dev/cdrom
```

or

```
# mount -r -t iso9660 /dev/cdrom /mnt/cdrom
```

Once you have access to the CD, change directories to the CD and run the installer.

5.6

Running the Installer

Root permission is required to install RPMs or the Subscription Management Server (used with a floating subscription). Gain root permission before starting the Installer if you intend to install RPMs or the Subscription Management server.

NOTE: You cannot use the Installer to install software to directories for which you do not have access or permission.

To run the installer, change into a directory that contains RPMs, a `tar` file, or a mounted ISO image or CD for the compiler environment that you want to install. Then, run the installer.

```
$ cd pathscale-eko-compilers-<version>
$ ./install
```

If you are using an ISO image or CD use these commands:

```
$ cd <mount_point>
$ ./install
```

A screen similar to the one below will prompt you through the install process.

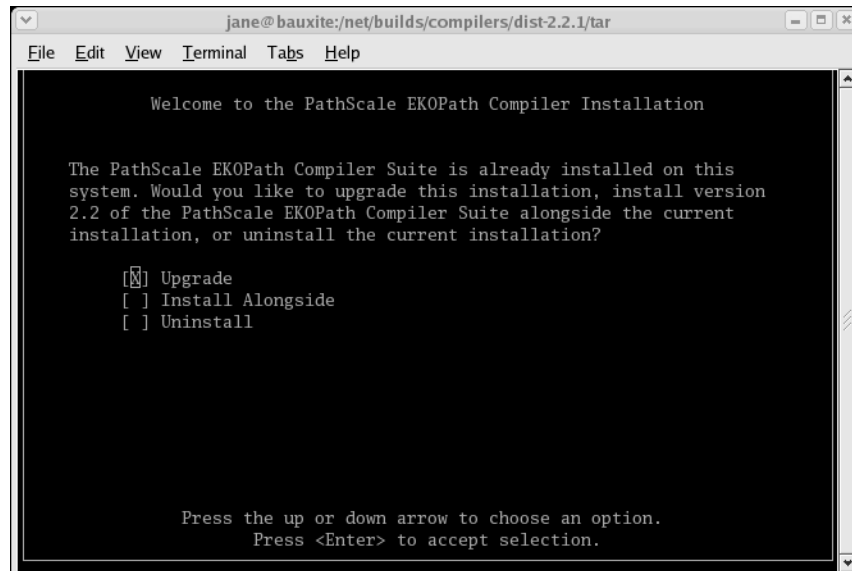


Figure 5-1. Installer Welcome Screen

5.7 **What the Installer is Doing**

The installer first checks to see what version(s) of the Compiler Suite are already installed on the system, and what access and permissions you have for installation. Then it will check for and configure your subscription file.

The Installer will also notify you of suggested configuration changes to your `LD_LIBRARY_PATH` environment variable. If you are doing an RPM install, the Installer will make those changes for you.

5.8 **Exiting the Installer**

The Installer will automatically exit after completing the installation. If you type `Control-c` before entering the Installation Progress screen, you will exit the Installer without making any changes. If you exit during or after this screen, you will have a partial or incomplete installation. To remove an unwanted installation, re-run the Installer and select "Uninstall" on the first screen.

5.9 **Use the Installer to Upgrade or Uninstall Compilers**

If you select upgrade, you will be presented with a selection of available install options (tar, RPM). Choose the option that matches the type of installation you have already installed (tar, RPM). All previous instances of the compiler will be removed and the new version will be installed. Your existing subscription file will be moved to the new install location with the upgrade.

The PathScale Installer can also be used to uninstall existing PathScale compiler environments. When the Installer finds an existing PathScale compiler installation, it asks if you want to uninstall this version or install the newer version alongside. If you select "Uninstall" the Installer will remove the existing compiler installation.

Section 6

Uninstalling the Compiler

This section covers uninstalling both RPMs and non-root `tar` files, for both single version and multiple version installations.

6.1

Uninstalling compiler RPMs

To uninstall the PathScale compiler RPMs you will need to have root privileges.

If you have only one version of the compiler installed, and you want to uninstall it, do the following.

Using a `bash` shell, gain root privileges and type:

```
# rpm -e $(rpm -qa | grep '_psc$')
```

Using a `csch` shell, gain root privileges and type:

```
# rpm -qa | grep '_psc$' | xargs -r rpm -ev
```

You will need to remove all of the packages to uninstall the compiler.

NOTE: This process will uninstall all copies of the compiler, even if multiple versions are installed.

You may need to remove the directory where the software was installed (for example `/opt/pathscale` or your `<install_directory>`) after removing the RPMs.

To complete the uninstall for all versions of the compiler, run the following command (as root):

```
# rm /etc/profile.d/pathscale-compilers.*
```

This will remove compiler-related pathnames from the `$PATH` variable. If you use the installer tool, this step does not need to be done, as it is taken care of automatically. [See section 5.6](#) for more information.

6.1.1

Uninstalling One of Multiple Compiler Versions of RPMs

If you have more than one version of the compiler RPMs installed, say versions 3.2 and 2.5, and you want to remove version 2.5, type:

```
# rpm -qa | egrep "pathscale.*-2.5-.*psc" | xargs rpm -ev
```

In this command, replace "2.5" with the version you want to uninstall.

After removing the RPMs, re-link the commands for invoking the compiler by doing the following (replace the "3.2" in the command line with the version number that you are keeping):

```
# cd /opt/pathscale/bin
# ln -f pathcc-3.2 pathcc
# ln -f pathCC-3.2 pathCC
# ln -f pathf95-3.2 pathf95
```

6.2

Uninstalling compiler using Debian Packages

To uninstall the PathScale compiler RPMs you will need to have root privileges.

If you have only one version of the compiler installed, and you want to uninstall it, do the following.

```
# dpkg --purge `dpkg -l | grep pathscale | awk '{print $2}'`
```

You will need to remove all of the packages to uninstall the compiler.

NOTE: This process will uninstall all copies of the compiler, even if multiple versions are installed.

You may need to remove the directory where the software was installed (for example `/opt/pathscale` or your `<install_directory>`) after removing the RPMs.

To complete the uninstall for all versions of the compiler, run the following command (as root):

```
# rm /etc/profile.d/pathscale-compilers.*
```

This will remove compiler-related pathnames from the `$PATH` variable.

6.3

Uninstalling Compilers from a Non-root tar File

If you installed the compiler suite from a non-root `tar` file, and you have only one version of the compiler installed, just delete the entire install directory to uninstall the PathScale compilers. For example, if you installed the compilers in the default directory, type:

```
$ rm -rf /opt/pathscale
```

NOTE: If you uninstall a non-root `tar` file release of the compilers, delete the lines you added in [section 3.12](#) from `/etc/ld.so.conf` and rerun `ldconfig`.

6.3.1

Uninstalling One of Multiple Compiler Versions of Non-root tar Files

To uninstall a non-root `tar` file installation of the compiler where you have multiple versions of the compiler installed, locate the root directory of the installation. For example, if you have two versions installed into `/home/janedoe`, you would have two directories named something like this:

```
/home/janedoe/pathscale-compilers-suse9.0-3.2-122.2140
/home/janedoe/pathscale-compilers-suse9.0-3.2-20.2125
```

To remove the first directory use this command:

```
$ rm -rf /home/janedoe/pathscale-compilers-suse9.0-3.2-122.2140
```

The other installation, /home/janedoe/pathscale-compilers-suse9.0-3.2-20.2125, will be unaffected.

Remember to remove any reference to an uninstalled compiler from /etc/ld.so.conf, and rerun ldconfig (see [section 3.12](#) for more information on ldconfig).

NOTE: To uninstall a non-root tar file, you need to be the same user (with the same privileges) who installed the non-root tar file.

Appendix A

Subscription Management Details

The subscription management software includes components that verify your subscription and help you to track your usage of the PathScale software and tools. Subscription management is required for the PathScale Compiler Suite and the PathScale debugger.

This section explains more about the subscription management model.

A.1

Modes of Operation

The subscription management software supports three modes of operation:

- **Nodelocked** single user mode - A single named user on a single machine. For this mode you will need a subscription file residing on the machine running the QILogic PathScale software.
- **Nodelocked** multiple user mode - Similar to single user mode. A list of authorized usernames is required.
- **Floating** mode - Multiple users on multiple machines For this mode you will need a subscription file and a running Subscription Manager server located on some (accessible) machine in your network

All of these types of subscriptions make use of *subscription management client software*, which is installed automatically as a part of the compiler or debugger software.

The subscription management client software requires a *valid subscription file*. This file is not supplied with the software distribution, but is obtained separately from PathScale.

A.2

Nodelocked Single User

This subscription type allows a single user to run the compilers on a single machine. The information needed by PathScale in order to issue a subscription of this type is:

1. The MAC address of the machine upon which the compiler will run
2. The username of the user who will run the compiler

The subscription allows the named user to run the compiler an unlimited number of times on the specified host. The subscription file is tied to the host and cannot be used on any other host.

If the machine to which the subscription is locked contains multiple active Ethernet interfaces, the MAC address passed to PathScale must match one of them. If we lock to an interface that is not marked as UP, then the node locking will fail.

A.3

Nodelocked, Multiple User

This subscription is very similar to the Nodelocked, single user subscription in that it allows a number of users to run the compiler on a single machine. Like the single-user case, the list of usernames for the authorized users is required, along with the MAC address of the host in order to issue the subscription.

A.4

Floating

This is the most flexible of the subscription schemes and is most useful for a larger group. Like other commercial licensing schemes, the floating subscription scheme allows multiple users to run the compilers on multiple machines up to the limit of 'seats'. In order to use this subscription scheme, it is necessary to run a daemon on your network that is accessible from the machines running the compiler.

The mainstay of the floating subscription model is the concept of a 'seat'. This is defined to be a unique username and hostname pair. When you buy a floating subscription, PathScale needs to know how many seats you will require. The subscription daemon keeps track of the number of seats you have used at any time and how many are available.

Seats are issued in the form of a 'lease' for a fixed period of time. When you run the compiler, the daemon will verify that there is a seat available for you and issue a lease tied to the username of the person running the compiler and the hostname of the machine upon which the compiler is running. For the duration of the lease, the username/hostname pair has access to the compiler to run an unlimited number of times. If a leaseholder runs the compiler within the lease period, the lease is automatically renewed.

It should be noted that this is different from the traditional licensing model used by other licensing software. Whereas the PathScale model issues a user a lease for a period of time, the traditional model is to track the number of concurrent users, each user blocking until a seat is available.

The subscription daemon uses the following formula to determine if there are seats available:

$$\text{numseats} = \max(\text{users}, \text{machines})$$

For example, if a site has 10 users and 20 machines, and the subscription specifies that the number of seats is 10, then 10 users can execute on 10 or less machines.

Like the Nodelocked subscription, the floating subscription daemon is tied to a single host, specified by the MAC address. This means that the daemon cannot be run on another host. If this is required, another subscription file must be generated.

When a user runs the compiler that has a floating subscription, the compiler must be able to find the daemon. The following algorithm is used:

1. First see if it can find the subscription file. It searches for the a file called pscsubscription.xml in the following places:
 - a. The contents of the environment variable
PATHSCALE_SUBSCRIPTION_DIR
 - b. The user's home directory
 - c. The location in which the compiler was installed
2. If the subscription file was found, look inside it to see if the <daemon/> tag was specified. If so, use this to determine the host and port of the server
3. If the subscription file was not found, look for the server on the following addresses:
 - a. The contents of the environment variable
PATHSCALE_SUBSCRIPTION_DAEMON
 - b. The hostname 'pscsubscriptionserver' (if it exists)

The floating subscription daemon runs by default on port TCP/7143, but this can be changed when the subscription was issued. It is not possible to change the port number after the subscription file is generated without generating a new file. If the port number is changed from the default, the compiler must be told what the new port number is. This is done either by allowing the compiler to see the subscription file or by specifying the hostname and port in the PATHSCALE_SUBSCRIPTION_DAEMON environment variable.

A.4.1

Named Users for Floating Subscription

A floating subscription can be tied to a named set of users. Only the named users will be granted leases for available seats. No other users will be allowed to use the system. The administrator can modify the file to add or remove users.

A.5

Subscription File

This file is a plain text file that contains information about the subscription. It is not supplied on the CD, but is emailed to you after the relevant information about the subscribed systems is obtained. The file is in an XML-like format and can be edited by the administrator. Beware that most of the fields in the file are subject to a checksum and any modification to these fields will invalidate the subscription.

For floating subscriptions only, the following fields are editable by the administrator:

```
<users value="" exclusive="" />
```

This is the list of users and the exclusivity setting. The list of users is a comma-separated string of usernames (no spaces allowed). The value of 'exclusive' is either "true" or "false". For example:

```
<users value="dallison,joeb,alice1,bob" exclusive="false"/>
<daemon host="" port=""/>
```

This specifies where the daemon runs. The 'host' is the hostname of the machine. The 'port' is the TCP port number. Normally you would not edit this field. For example:

```
<daemon host="12.4.5.6" port="7146"/>
```

Depending on the subscription model, the subscription file must be available to certain components of the subscription software:

- Nodelocked: The file must be readable by the user running the compiler.
- Floating: The file must be readable by the subscription server (daemon) and possibly the user running the compiler.

In the case of a floating subscription, the file may contain information about where to locate daemon so it might be useful for the users to be able to see it.

NOTE: The mode (nodelocked or floating) is determined by the PathScale subscription management file (`psscsubscription-<product>*.xml`) that you received from PathScale. Please contact PathScale if you wish to change your subscription mode.

A.6

Perpetual Right-to-use

The license with perpetual right-to-use enables you to indefinitely run any version of the licensed software released during your paid support period. Software released after your paid support period is over will not run with that license. Extensions to the support period may be purchased from PathScale. Contact support@pathscale.com for details.

A.7

Overview of Subscription Manager Client-Server Interaction

When using a floating subscription, some system administrators have registered concerns about network traffic and security. This section is a brief overview of the interaction between the client and server to help address those concerns.

The PathScale subscription management has been designed to make efficient use of the network. By default, the client and server communicate over port 7143. On the server side, the port used can be changed when the subscription file is requested from PathScale. On the client side, the port can be changed by specifying an alternate port number when defining the `PATHSCALE_SUBSCRIPTION_DAEMON` environment variable. The client and server must be using the same port numbers.

When the PathScale software initially starts up, it queries the subscription server providing the user name and system name to the server in a signed request. If a subscription is available, the server will then acknowledge the request with a signed reply containing a lease. If a subscription is not available, then the server will respond accordingly and the client will terminate with an appropriate error message.

The lease provided by the server consists of several components including the user name, the system name, a lease interval, and a lease duration. When the client receives a lease, it is cached in a `./pathscale` subdirectory of the user's home directory.

Every time a client is invoked, it will first check for a current lease in the cache. If the the lease interval has expired, then the client will contact the server and request an extension to the lease. If the client has not been able to contact the server within the duration of the lease, then the lease is no longer valid and the client will terminate with an appropriate error message. Both the lease duration and interval are defined by PathScale and cannot be modified by the user.

The client can be configured to wait instead of terminating by setting the environment variable `PSC_WAIT4COMPILER` to N, which tells the client to wait up to N minutes for a lease. If the variable is not set or is set to 0, the client will not wait and terminate.

If for any reason the cache directory cannot be created or is not readable, then the client will request a lease upon each invocation of the PathScale software. The server will handle this request as a request for a new lease if one does not exist, or a request for renewal of an existing lease if one already exists for that user on that system.

Appendix B

Troubleshooting

This section covers specific examples of issues you may encounter while using the PathScale compilers.

Support is available from support@pathscale.com. Also see the PathScale web site for information on the latest releases.

<http://www.pathscale.com/support/>

For information on how to submit a bug, see the *PathScale Compiler Suite Support Guide*.

B.1

Installation Problems

B.1.1

Problem. Trouble Installing on SLES or SUSE 9.x

Error message while installing SLES or SUSE 9.x RPMs.

Symptom. You type:

```
# rpm -Uvh *.rpm
```

Error message:

```
error: failed dependencies:
    libbfd-2.12.90.0.15.so is needed by
    binutils-2.14.90.0.5-25.19_psc
    pathscale-compilers-common < 1.2 conflicts with
    pathscale-subscription-client-sles-1.2-28.23_psc
```

Solution. Install using the “`--nodeps`” argument. This works around the dependencies problems with some SLES or SUSE 9.x RPMs.

Type:

```
# rpm -Uvh --nodeps *.rpm
```

B.1.2

Problem: Installing SUSE/SLES, Non-default Location

You may have trouble upgrading to a new release of the EKOpath compilers on supported SUSE and SLES systems where you have performed a relocated installation of the compilers (installed in a non-default directory). You will need to uninstall the previous version of the EKOpath compilers before installing the new version. Refer to [section 6.1](#) for information on uninstalling the compiler.

Symptom. You type:

```
# rpm -Uvh --prefix <install_directory> <install_options>
```

The install does not seem to work correctly. You notice that many files seem to be missing; especially `pathcc`, `pathCC`, all the documents, all of the headers.

Solution:

You need to uninstall the previous version of the EKOp`ath` compilers before installing the new version. Refer to [section 6.1](#) for information on uninstalling the compiler.

B.1.3

Problem: Failed Dependencies

Trying to install a package on a version of Linux (Red Hat 7.2) that is not new enough to run the compiler.

Symptom. You type:

```
$ rpm -Uvh pathscale-f90-3.2-234.2865_rhel3_psc.i386.rpm
```

Error message:

```
error: failed dependencies:  
libc.so.6(GLIBC_2.3) is needed by  
pathscale-compilers-common
```

Solution. Upgrade to a newer version of Linux. [See section 2.1](#) for information on supported distributions.

B.1.4

Problem: Missing Packages

Trying to install the compiler package on a supported system, but without some of its companion packages installed.

Symptom. You type:

```
$ rpm -Uvh pathscale-f90-2.2.99-234.2865_rhel3_psc.i386.rpm
```

Error message:

```
error: Failed dependencies:  
pathscale-compilers-common-2.2.99 is needed by  
pathscale-f90
```

Solution. Install the additional required packages. Refer to [section 3.5](#) for installation instructions. See [appendix D](#) for the lists of package dependencies for the different platforms.

B.1.5

Problem: Removing Dependent Packages

Trying to remove a package that other packages depend on.

Symptom. You type:

```
$ rpm -e pathscale-compilers-common
```

Error message:

```
error: Failed dependencies:
  pathscale-compilers-common = 2.1-257.2867_fc1_psc is needed
  by (installed) pathscale-c-2.1-257.2867_fc1_psc.i386
  pathscale-compilers-common = 2.1-257.2867_fc1_psc is needed
  by (installed) pathscale-c++-2.1-257.2867_fc1_psc.i386
  pathscale-compilers-common = 2.1-257.2867_fc1_psc is needed
  by (installed) pathscale-compilers-docs-2.1-257.
  2867_fc1_psc.i386
  pathscale-compilers-common = 2.1-257.2867_fc1_psc is needed
  by (installed) pathscale-compilers-libs-lgpl-2.1-257.
  2867_fc1_psc.i386
  pathscale-compilers-common = 2.1-257.2867_fc1_psc is needed
  by (installed) pathscale-f90-2.1-257.2867_fc1_psc.i386
```

Solution. You can't remove packages with dependencies, without removing the dependent packages at the same time. If you want to remove a package with dependencies, remove it and all its dependent packages at once. Refer to [section 6](#) for instructions on how to remove the compiler. You can also refer to [appendix D](#) for information on package dependencies.

B.2

Subscription Problems

If you are having trouble with the Subscription Manager, use the `-subverbose` option with the compilers. The information from this option is useful for diagnosing problems. You need to do an actual compile with the `-subverbose` option to see the output.

For example, you might type:

```
$ pathcc -subverbose hello.c
```

See the *PathScale Subscription Management User Guide* for more information, sample output, and troubleshooting tips.

B.2.1

Can't Find Subscription File

If you receive the following message, either the subscription is not installed properly or the environment is not completely configured.

```
"No subscription file found. The PathScale compiler cannot run
without a subscription.
Please see http://www.pathscale.com/subscription/1.1/messages.html
for details"
```

B.2.2

Misconfigured Environment

There is the potential for a mis-configured environment to send the user name and host name to a system that is not the server. The two circumstances are:

- ? The `PATHSCALE_SUBSCRIPTION_DAEMON` environment variable has the wrong hostname in it.
- ? The name server resolves the hostname `pscsubscriptionserver` to a system that is not the correct system.

Both of these cases will behave just like any other misdirected network request and most likely will result in the client terminating with a "subscription unavailable" error message due to the lack of an appropriate response.

B.2.3

Getting Information on Subscription Problems

If you are having trouble with PathScale subscription management while using the compilers, use the `-subverbose` option as you compile. The information from this option is useful for diagnosing problems. You will need to do an actual compile with the `-subverbose` option to see the output.

For example, you might type:

```
$ pathcc -subverbose hello.c
```

Sample output:

```
$ pathcc -subverbose hello.c
Subscription client: looking for /opt/pathscale/lib/2.3/subclient
Subscription client: No PATHSCALE_SUBSCRIPTION_DAEMON variable is
present
Subscription client: looking for pscsubscriptionserver in DNS
Subscription client: trying CNAME server: pscsubscriptionserver
Subscription client: pscsubscriptionserver resolved to
192.168.3.181
Subscription client: no lease file, forcing server contact
Subscription client: attempting to contact subscription server at
pscsubscriptionserver: 7143
Subscription client: subscription granted by server
```

B.2.4

Problems Starting the Subscription Server

If you are having problems starting the subscription server, here are some things to try. First you need to verify that the subscription server is running. To do this, look for a process running on the system with the name `subserver`:

```
$ ps -ef | grep subserver
```

If the server is running, you should see something like this on the terminal:

```
dave 12730      1   0 Apr21 ?          00:00:07 subserver -d
dave 28897    28873   0 15:38 pts/3    00:00:00 grep subserver
```

If you only see the following (or nothing at all), then the server is not running:

```
dave 28897    28873   0 15:38 pts/3    00:00:00 grep subserver
```

B.2.5

Server is Running But the Compiler Can't Find It

The server is a TCP listening program that listens, by default, to all the interface addresses available to it. Some installations prefer the server to listen on a specific port number. This port number is specified in your `pscsubscription-<product>-* .xml` file in the `<daemon/>` tag. For example, if the `pscsubscription-<product>-* .xml` file has something like the following in it:

```
<daemon host="foo" port="8654"/>
```

then the server will run on the host named 'foo' on port 8654. If the line is missing from the file or the host and port attributes are blank, then the server will be running on all available interfaces.

NOTE: If you need to change the host or port that the server runs on, please contact PathScale to have a new subscription file generated that includes the updated information.

In order to verify that the server is running on the correct port, type the following command:

```
$ netstat -l | grep 7143
```

(Replace the 7143 with the appropriate port number from the `pscsubscription-<product>-* .xml` file if necessary.)

The output will be something like this:

```
tcp      0    0 quarry.internalke:7143  *:*      LISTEN
```

If there is no output then the server is not running on the specified port.

If you have verified that the server is running on the required port, you should try running the compiler (or other software) with the `-subverbose` option.

NOTE: You will have to actually compile a file for the `-subverbose` option to work.

If all is well, the following output will appear:

```
$ pathcc -subverbose hello.c
Subscription client: looking for /opt/pathscale/lib/2.3/subclient
Subscription client: No PATHSCALE_SUBSCRIPTION_DAEMON variable is present
Subscription client: looking for pscsubscriptionsserver in DNS
Subscription client: trying CNAME server: pscsubscriptionsserver
Subscription client: pscsubscriptionsserver resolved to 192.168.3.181
Subscription client: no lease file, forcing server contact
Subscription client: attempting to contact subscription server at
pscsubscriptionsserver: 7143
Subscription client: subscription granted by server
```

If things are not going well, then you might see the following output:

```
Subscription client: looking for /opt/pathscale/lib/2.3/subclient
Subscription client: Reading subscription file:
/opt/pathscale/lib/2 . 3/pscsubscription-compilers-2082 .xml
Subscription client: Trying to contact server specified in subscription
file /opt/pathscale/lib/2 . 3/pscsubscription-compilers-2082 .xml:
Subscription client: trying server set in subscription file <daemon/>
entry: tourmaline:7 143
Subscription client: tourmaline resolved to 192.168.2.30
Subscription client: no lease file, forcing server contact
Subscription client: attempting to contact subscription server at
tourmaline:7143
Subscription client: direct connection to server failed: No route to host
Subscription client: failed to connect to subscription server
Subscription client: *** Subscription: unable to get subscription due to
daemon error: No route to host
*** Subscription: Unable to find a server. The PathScale products cannot
run without a subscription. Please see
http://www.pathscale.com/subscription/1.1/msgs.html for details.
Subscription client: The following directories were searched for the
file: Subscription client:/home/username/ .pathscale
Subscription client:/opt/pathscale/lib/2 . 3
```

In this case the client found a <daemon/> entry in the subscription file that pointed to a host (tourmaline on port 7144). It is likely that the server is not listening on that port. To verify this, use the `netstat -l` command to see if port 7144 on the server is open and listening:

```
$ netstat -l | grep 7144
[no output]
```

The fact that we received no output from the `netstat` command tells us that the port is not open and thus there is no server running on that port.

The client looks for the server using the following process (in the order specified):

1. Look for <daemon/> tag in `pscsubscription-<product>-* .xml` file
2. Look in `PATHSCALE_SUBSCRIPTION_DAEMON` environment variable
3. Look for the hostname `pscsubscriptionsserver`

B.2.6

Server is Not Running

If the server is not running, then it must have failed to start. First verify that you actually did type the command to start it (see section 3.17.5), and that it started up successfully. If that step was performed and the server is still not running, the next step is to look at the output from the server.

When running in daemon mode (with the `-d` option), the server writes output to the `/var/log/messages` file on the system using the `syslog ()` system call. The contents of this file can be seen using the following command (as root):

```
# cat /var/log/messages
```

This command will produce lots of messages. If the server was started recently, its output will be near the end of the `messages` file. If there are no obvious errors in the message output, and the server is still not running, you should try running the server by hand.

Type the following command, replacing the parts in angle brackets with the correct names for your configuration and environment. You do not want root privileges for this.

```
$ <install-path>/pathscale/lib/<version>/subserver -v --v -check
```

This command tells the server to run in verbose and debug mode. If the server runs successfully, the response should look something like this:

```
Using subscription file: /home/dave/pscsubscription-
compilers-2082 .xml using specified daemon address:
tourmaline
PathScale Subscription Server started on tourmaline:7546
  at Mon Oct 3 15:59:49 PDT 2005
lease manager initialized with 1 seats for subscription
file version 2.3
```

If the server fails to start, it will print information saying why it is not starting. If you don't understand the information from the server, please contact PathScale support at support@pathscale.com.

B.2.7

Problem: Server Fails to Start

The message is:

```
“ERROR: Unable to run subscription server - not authorized to execute on this node”
```

Reason: The MAC address supplied to PathScale does not match that of the machine on which the server is starting.

Solution: This may have a number of causes. First, check that the `/etc/hosts` file does not have an entry for the hostname of the machine under the loopback interface (`127.0.0.1`). It is common for some machines to be set up this way and it is incorrect. The only entry should be for the actual IP address of the machine. You can verify the IP address by using the `'ping'` command.

If the `/etc/hosts` file is correct, it may be that the machine has more than one Ethernet interface. In this case, you can use the `-iface` option to the `subserve r` command to change the interface on which the server runs. This will entail editing the server startup script in `/etc/init.d` and adding the `-iface` option to the `subserver` command.

If the `hosts` file is correct and there is only one interface, then ensure that the interface is running. This can be found using the `/sbin/ifconfig -a` command and seeing if the 'UP' flag is present for the appropriate interface.

In addition, check [section 3.18](#) to verify that you have installed and configured the subscription file and/or Subscription Manager correctly.

B.2.8

Problem: Compiler says "Unable to obtain subscription" and then Stops

Symptom: The compiler prints this message on startup and refers to the PathScale website. The compilation is then aborted.

Solution: This message means that the compiler was unable to find the Subscription Manager software. This is a program called 'subclient' that resides in the same location as the compiler software (`/opt/pathscale/lib/<version>` by default).

Check [section 3.18](#) to verify that you have installed and configured the subscription file and/or Subscription Manager correctly.

B.2.9

Problem: Compiler: "Bad subscription version: <text>" and then Stops

Symptom: The compiler prints this message on startup and refers to a PathScale website. The compilation is then aborted.

Solution: This message means that the version of the subscription client was incorrect for this release of the compiler. Please verify that the correct versions of both the compiler and subscription software have been installed.

Check [section 3.18](#) to verify that you have installed and configured the subscription file and/or Subscription Manager correctly.

B.2.10

Problem: Compiler: "No subscription file found" and then stops

Symptom: The compiler prints this message on startup and refers to a PathScale website. The compilation is then aborted.

Solution: This means that the subscription file was not found by the subscription management software. This file is called 'pscsubscription-<product>-* .xml' and is usually located in the same location as the subscription software (`/opt/pathscale/lib/<version>` by default). If the file is located elsewhere on your system, set the environment variable `PATHSCALE_SUBSCRIPTION_DIR` to the directory in which the file is located.

The subscription file is not available on the installation CD or with the download `tar` file of RPMs, but is sent separately to you from PathScale. See [section 3.3.2](#) for information on obtaining a subscription file from PathScale. Check [section 3.18](#) to verify that you have installed and configured the subscription file and/or Subscription Manager server correctly.

B.2.11

Problem: Unable to Upgrade with Current Subscription:

The message is:

```
*** Subscription: Unable to obtain subscription.
```

Symptom: The compiler prints this message on startup and refers to a PathScale website. The compilation is then aborted. The subscription file worked before the upgrade.

Solution: A valid subscription file was not found by the Subscription Manager. If you know you have a valid subscription, and you have upgraded your compiler to a new release, you will need to copy the `pscssubscription-<product>-* .xml` file to an additional location.

Copy the subscription file from:

```
<install_dir>/lib/<old_version>/pscssubscription-<product>-* .xml
```

Into this new location:

```
<install_dir>/lib/<new_version>/pscssubscription-<product>-* .xml
```

B.2.12

Problem: Compiler prints lines of the form: * Subscription: <text>**

Symptom: These are warnings from the subscription system. They are for information only, but are important because they can signal a problem with the subscription. Problems could be:

- ? Subscription expired
- ? Too many users or hosts

The output will look something like this:

```
$ pathcc -o hello hello.c
*** Subscription: Unable to obtain a subscription. The PathScale
products cannot run without a subscription.
Please see http://www.pathscale.com/subscription/1.1/
msgs.html for details.
```

Solution: These messages are informational only. The compilation is not affected, unless another error occurs that prevents the compiler from continuing. Check [section 3.18](#) to verify that you have installed and configured the subscription file and/or Subscription Manager correctly.

B.2.13

Problem: Ethernet Port is Down

The ethernet port with the subscription MAC address is down. This often occurs on a laptop computer. This can occur if no IP address is present

The error will look something like this:

```
Subscription client: looking for usr/local/pathscale/lib/2.3/  
subclient  
Subscription client: Using subscription file: /usr/local/  
pathscale/lib/2.3/pscsubscription-compilers-2082.xml Subscription  
client: chose subscription file version 2.2  
*** Subscription: not authorized to execute on this node  
Compilation terminated
```

Solution: Run `/sbin/ifconfig -a` and check the MAC address, IP address, and status of the IP port. Bring the IP port up, if necessary.

B.2.14

Problem: Nodelocked Perpetual Subscription Fails

Symptom: You have a nodelocked perpetual subscription and you cannot run your software.

If a nodelocked perpetual subscription fails, the error message is:

```
*** Subscription: Your subscription does not allow the  
use of this version of the product. Under your agreement,  
you are allowed to run products released before the  
expiration of your subscription indefinitely. This product  
was released on Tuesday October 18 10:55:14 2005 which is  
beyond that date. You mzy either run older versions of this  
product or you may contact PathScale and formally renew your  
support agreement.
```

Solution: Either run older versions of the software or contact PathScale to renew your support agreement.

B.2.15

Problem: Floating Perpetual Subscription Fails

Symptom: You have a floating perpetual right-to-use subscription and you cannot run your software.

If a floating perpetual subscription fails, the error message is:

```
*** Subscription: Your subscription does not allow the  
use of this version of the product. Under your agreement,  
you are allowed to run products released before the  
expiration of your subscription indefinitely. This product  
was released on Fri Jun 10 12:12:22 2005 which is beyond that  
date. You may either run older versions of this product or  
you may contact PathScale and formally renew your support  
agreement.
```

Solution: Either run older versions of the software or contact PathScale to renew your support agreement.

B.2.16

Problem: Subscription Server Hangs with ROCKS

Symptom: The subscription server can hang forever when using ROCKS

Here is a hint for anyone who uses ROCKS (an unsupported operating system).

The subscription server will hang forever with this message:

```
[root@myserver subscription-server] # /etc/rc.d/init.d/pathscale-  
sub start  
Checking for PathScale compiler subscription: [ OK ]  
Starting PathScale compiler subscription server:  
Jul 6 10:54:38 myserver runuser: Enter file in which to save the  
key  
    (/var/log/pathscale-sub/.ssh/id_rsa):  
Jul 6 10:56:02 csebeo runuser:  
    /var/log/pathscale-sub/.ssh/id_rsa.pub  
Jul 6 10:56:02 csebeo runuser:  
    /var/log/pathscale-sub/.ssh/id_rsa
```

To fix this, do the following:

```
$ su - psc_sub  
<hit return a few times to take the default keys>
```

B.3

Other Problems

B.3.1

NLSPATH Causes Error

The NLSPATH environment variable is not set to include the location of the compiler error messages.

Symptom. The compiler exits with a message like:

```
pathf95 INTERNAL: Unable to open message system.
```

Solution. Either unset the NLSPATH environment variable or set it to point to the location of the error messages, using %N for the base name of the file. Modify your NLSPATH to read something like this:

```
<old_nls_path>:<eko_install_directory>/lib/3.2/%N.cat
```

where `old_nls_path` is the previous value of this variable, and `eko_install_directory` is the directory where you installed the PathScale compilers (usually `/opt/pathscale`). For example, if the compiler version is 3.2, set NLSPATH to:

```
/opt/pathscale/lib/3.2/%N.cat
```


Appendix C

Glossary

This section describes common terms that are used in this manual.

Table 6-1.

<i>CD image</i>	Either the ISO image or the physical CD distribution of the PathScale Compiler Suite.
<i>daemon</i>	A program or process running in the background that checks the status of the subscription.
<i>download tar file</i>	The PathScale Compiler Suite distribution available from the PathScale website.
<i>exclusive users</i>	A mode used with the floating subscription. Exclusive users are named in the subscription file (<code>psscsubscription-<product>-* .xml</code>). When this mode is used by the Subscription Manager, only named users will be granted leases to use the compiler.
<i>floating subscription</i>	One of the two types of subscription management available. It requires that Subscription Manager server software be installed and correctly configured, enabling users to access the compilers from different machines.
<i>hostname</i>	The unique name that identifies a computer on a network, used to identify it to other computers or processes on the network.
<i>install directory</i>	The directory where you installed the PathScale Compiler Suite. This is <code>/opt/pathscale</code> by default.
<i>ISO image</i>	Disk image of an ISO 9660 file. In addition to the data, filesystem metadata is contained. May be mounted like a physical disk.
<i>lease</i>	A subscription to use the compiler for a fixed period of time. Lease availability is varied by the subscription daemon.
<i>MAC address</i>	Media Access Control Address. Used by the subscription daemon to determine and assign leases. A lease is tied to a single host, specified by the MAC address, in either the nodelocked or floating subscriptions.

Table 6-1.

<i>nodelocked subscription</i>	One of the two types of subscription management available . Nodelocked subscriptions do not require the PathScale Subscription Manager server software to be installed.
<i>non-exclusive users</i>	A mode used with the floating subscription used by the Subscription Manager. Non-exclusive users can be named in the subscription file (<code>psscsubscription-<product>-* .xml</code>), but other users will be allowed to use the compiler, based on the availability of leases.
<i>perpetual right-to-use</i>	A subscription that enables you to run software indefinitely, but restricts you from upgrading to software newer than that available during your paid subscription.
<i>PathScale subscription management</i>	The software included with the PathScale Compiler Suite that helps you track your usage of the compilers. It requires valid subscription file that species the number of leases you have purchased.
<i>PathScale Subscription Manager server</i>	Separate server software used with a floating subscription. This server needs to be installed and congured on a system accessible to the subscription client software.
<i>RPMs</i>	(Red Hat Package Manager) This refers to both the format and the tools for verifying, installing, uninstalliing, and updating software packages.
<i>subscription file</i>	The plain text file that contains information about your subscription (<code>psscsubscription-<product>-* .xml</code>). For floating subscriptions, some of the fields are editable.

Appendix D

Package Dependencies

The following are lists of dependences, by package name, for each of the PathScale Compiler Suite's supported platforms. As a general rule, for your distribution, you need to install compatibility libraries, a 32-bit runtime, and python. The tables below show this information in greater detail.

D.1

Specific Packages by Platform s

The packages are given here as an example. The version number (here 3.2) for the PathScale packages changes with each release. Note that each set of RPMs uses a build identifier xxx and a distribution identifier yyy. The architecture in the RPM name is designated by x86_64 or i386.

The required base compiler packages and their dependencies are listed first. Optional packages and their dependencies (if required) follow.

D.1.1

Red Hat Enterprise Linux Workstation 4

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the Red Hat Enterprise Linux Workstation 4 platform.

Release: Red Hat Enterprise Linux WS release 4 (Nahant)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.0	x86_64-redhat-linux-gnu
binutils	x86_64	2.15.92.0.2	x86_64-redhat-linux
glibc-devel	i386	2.3.4	i386-redhat-linux-gnu
glibc-devel	x86_64	2.3.4	x86_64-redhat-linux-gnu
glibc	i686	2.3.4	i686-redhat-linux-gnu
glibc	x86_64	2.3.4	x86_64-redhat-linux-gnu
glibc-profile ^a	i686	2.3.4	i686-redhat-linux-gnu
glibc-profile ^a	x86_64	2.3.4	x86_64-redhat-linux-gnu
grep	x86_64	2.5.1	x86_64-redhat-linux-gnu
libgcc	i386	3.4.3	i386-redhat-linux
libstdc++	i386	3.4.3	i386-redhat-linux
make	x86_64	3.80	x86_64-redhat-linux-gnu
python	x86_64	2.3.4	x86_64-redhat-linux-gnu

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	3.4.3	x86_64-redhat-linux
libstdc++-devel	x86_64	3.4.3	x86_64-redhat-linux
libstdc++	i386	3.4.3	i386-redhat-linux
libstdc++	x86_64	3.4.3	x86_64-redhat-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
ncurses	i386	5.4	i386-redhat-linux-gnu

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

Name	Arch	Version	Platform
ncurses	i386	5.4	i386-redhat-linux-gnu

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.2

Red Hat Enterprise Linux Workstation 5

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the Red Hat Enterprise Linux Workstation 5 platform.

Release: Red Hat Enterprise Linux WS release 5 (Tikanga)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.1	x86_64-redhat-linux-gnu
binutils	x86_64	2.17.52.0.6	x86_64-redhat-linux
glibc-devel	i386	2.5	i386-redhat-linux-gnu
glibc-devel	x86_64	2.5	x86_64-redhat-linux-gnu
glibc	i686	2.5	i686-redhat-linux-gnu
glibc	x86_64	2.5	x86_64-redhat-linux-gnu
glibc-profile ^a	i686	2.5	i686-redhat-linux-gnu
glibc-profile ^a	x86_64	2.5	x86_64-redhat-linux-gnu
grep	x86_64	2.5.1	x86_64-redhat-linux-gnu
libgcc	i386	4.1.1	i386-redhat-linux
libstdc++	i386	4.1.1	i386-redhat-linux
make	x86_64	3.81	x86_64-redhat-linux-gnu
python	x86_64	2.4.3	x86_64-redhat-linux-gnu

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.1.1	x86_64-redhat-linux
libstdc++-devel	x86_64	4.1.1	x86_64-redhat-linux
libstdc++	i386	4.1.1	i386-redhat-linux
libstdc++	x86_64	4.1.1	x86_64-redhat-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
ncurses	i386	5.5	i386-redhat-linux-gnu

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

Name	Arch	Version	Platform
ncurses	i386	5.5	i386-redhat-linux-gnu

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.3

Fedora Core 3

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the Fedora Core 3 platform.

Release: Fedora Core release 3 (Heidelberg)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.0	x86_64-redhat-linux-gnu
binutils	x86_64	2.15.92.0.2	x86_64-redhat-linux
libstdc++	i386	3.4.4	i386-redhat-linux
glibc-devel	i386	2.3.6	i386-redhat-linux-gnu
glibc-devel	x86_64	2.3.6	x86_64-redhat-linux-gnu
glibc	i686	2.3.6	i686-redhat-linux-gnu
glibc	x86_64	2.3.6	x86_64-redhat-linux-gnu
glibc-profile ^a	i686	2.3.6	i686-redhat-linux-gnu
glibc-profile ^a	x86_64	2.3.6	x86_64-redhat-linux-gnu
grep	x86_64	2.5.1	x86_64-redhat-linux-gnu
libgcc	i386	3.4.4	i386-redhat-linux
make	x86_64	3.80	x86_64-redhat-linux-gnu
python	x86_64	2.3.4	x86_64-redhat-linux-gnu

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	3.4.4	x86_64-redhat-linux
libstdc++-devel	x86_64	3.4.4	x86_64-redhat-linux
libstdc++	i386	3.4.4	i386-redhat-linux
libstdc++	x86_64	3.4.4	x86_64-redhat-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
ncurses	i386	5.4	i386-redhat-linux-gnu

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

Name	Arch	Version	Platform
ncurses	i386	5.4	i386-redhat-linux-gnu

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.4

Fedora Core 4

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the Fedora Core 4 platform.

Release: Fedora Core release 4 (Stentz)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.0	x86_64-redhat-linux-gnu
binutils	x86_64	2.15.94.0.2.2	x86_64-redhat-linux
compat-libstdc++-33	i386	3.2.3	i386-redhat-linux
glibc-devel	i386	2.3.6	i386-redhat-linux-gnu
glibc-devel	x86_64	2.3.6	x86_64-redhat-linux-gnu
glibc	i686	2.3.6	i686-redhat-linux-gnu
glibc	x86_64	2.3.6	x86_64-redhat-linux-gnu
glibc-profile ^a	i686	2.3.6	i686-redhat-linux-gnu
glibc-profile ^a	x86_64	2.3.6	x86_64-redhat-linux-gnu
grep	x86_64	2.5.1	x86_64-redhat-linux-gnu
libgcc	i386	4.0.2	i386-redhat-linux
make	x86_64	3.80	x86_64-redhat-linux-gnu
python	x86_64	2.4.3	x86_64-redhat-linux-gnu

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.0.2	x86_64-redhat-linux
libstdc++-devel	x86_64	4.0.2	x86_64-redhat-linux
libstdc++	i386	4.0.2	i386-redhat-linux
libstdc++	x86_64	4.0.2	x86_64-redhat-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
ncurses	i386	5.4	i386-redhat-linux-gnu

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

Name	Arch	Version	Platform
ncurses	i386	5.4	i386-redhat-linux-gnu

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.5

Fedora Core 5

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the Fedora Core 5 platform.

Release: Fedora Core release 5 (Bordeaux)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.1	x86_64-redhat-linux-gnu
binutils	x86_64	2.16.91.0.6	x86_64-redhat-linux
compat-libstdc++-33	i386	3.2.3	i386-redhat-linux
glibc-devel	i386	2.4	i386-redhat-linux-gnu
glibc-devel	x86_64	2.4	x86_64-redhat-linux-gnu
glibc	i686	2.4	i686-redhat-linux-gnu
glibc	x86_64	2.4	x86_64-redhat-linux-gnu
glibc-profile ^a	i686	2.4	i686-redhat-linux-gnu
glibc-profile ^a	x86_64	2.4	x86_64-redhat-linux-gnu
grep	x86_64	2.5.1	x86_64-redhat-linux-gnu
libgcc	i386	4.1.1	i386-redhat-linux
make	x86_64	3.80	x86_64-redhat-linux-gnu
python	x86_64	2.4.3	x86_64-redhat-linux-gnu

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.1.1	x86_64-redhat-linux
libstdc++-devel	i386	4.1.1	i386-redhat-linux
libstdc++-devel	x86_64	4.1.1	x86_64-redhat-linux
libstdc++	i386	4.1.1	i386-redhat-linux
libstdc++	x86_64	4.1.1	x86_64-redhat-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
ncurses	i386	5.5	i386-redhat-linux-gnu

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

Name	Arch	Version	Platform
ncurses	i386	5.5	i386-redhat-linux-gnu

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.6

Fedora Core 6

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the Fedora Core 6 platform.

Release: Fedora Core release 6 (Zod)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.1	x86_64-redhat-linux-gnu
binutils	x86_64	2.17.50.0.6	x86_64-redhat-linux
glibc-devel	i386	2.5	i386-redhat-linux-gnu
glibc-devel	x86_64	2.5	x86_64-redhat-linux-gnu
glibc	i686	2.5	i686-redhat-linux-gnu
glibc	x86_64	2.5	x86_64-redhat-linux-gnu
glibc-profile ^a	i686	2.5	i686-redhat-linux-gnu
glibc-profile ^a	x86_64	2.5	x86_64-redhat-linux-gnu
grep	x86_64	2.5.1	x86_64-redhat-linux-gnu
libgcc	i386	4.1.1	i386-redhat-linux
libstdc++-33	i386	4.1.1	i386-redhat-linux
make	x86_64	3.81	x86_64-redhat-linux-gnu
python	x86_64	2.4.4	x86_64-redhat-linux-gnu

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.1.1	x86_64-redhat-linux
libstdc++-devel	i386	4.1.1	i386-redhat-linux
libstdc++-devel	x86_64	4.1.1	x86_64-redhat-linux
libstdc++	i386	4.1.1	i386-redhat-linux
libstdc++	x86_64	4.1.1	x86_64-redhat-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
ncurses	i386	5.5	i386-redhat-linux-gnu

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

Name	Arch	Version	Platform
ncurses	i386	5.5	i386-redhat-linux-gnu

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.7

Fedora Core 7

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the Fedora Core 7 platform.

Release: Fedora Core release 7 (Moonshine)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.2	x86_64-redhat-linux-gnu
binutils	x86_64	2.17.50.0.12	x86_64-redhat-linux
glibc-devel	i386	2.6	i386-redhat-linux-gnu
glibc-devel	x86_64	2.6	x86_64-redhat-linux-gnu
glibc	i686	2.6	i686-redhat-linux-gnu
glibc	x86_64	2.6	x86_64-redhat-linux-gnu
glibc-profile ^a	i686	2.6	i686-redhat-linux-gnu
glibc-profile ^a	x86_64	2.6	x86_64-redhat-linux-gnu
grep	x86_64	2.5.1	x86_64-redhat-linux-gnu
libgcc	i386	4.1.2	i386-redhat-linux
libstdc++-33	i386	4.1.2	i386-redhat-linux
make	x86_64	3.81	x86_64-redhat-linux-gnu
python	x86_64	2.5	x86_64-redhat-linux-gnu

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.1.2	x86_64-redhat-linux
libstdc++-devel	i386	4.1.2	i386-redhat-linux
libstdc++-devel	x86_64	4.1.2	x86_64-redhat-linux
libstdc++	i386	4.1.2	i386-redhat-linux
libstdc++	x86_64	4.1.2	x86_64-redhat-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
ncurses	i386	5.6	i386-redhat-linux-gnu

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

Name	Arch	Version	Platform
ncurses	i386	5.6	i386-redhat-linux-gnu

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.8

Fedora Core 8

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the Fedora Core 8 platform.

Release: Fedora Core release 8 (Werewolf)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.2	x86_64-redhat-linux-gnu
binutils	x86_64	2.17.50.0.18	x86_64-redhat-linux
glibc-devel	i386	2.7	i386-redhat-linux-gnu
glibc-devel	x86_64	2.7	x86_64-redhat-linux-gnu
glibc	i686	2.7	i686-redhat-linux-gnu
glibc	x86_64	2.7	x86_64-redhat-linux-gnu
glibc-profile ^a	i686	2.7	i686-redhat-linux-gnu
glibc-profile ^a	x86_64	2.7	x86_64-redhat-linux-gnu
grep	x86_64	2.5.1	x86_64-redhat-linux-gnu
libgcc	i386	4.1.2	i386-redhat-linux
libstdc++-33	i386	4.1.2	i386-redhat-linux
make	x86_64	3.81	x86_64-redhat-linux-gnu
python	x86_64	2.5.1	x86_64-redhat-linux-gnu

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.1.2	x86_64-redhat-linux
libstdc++-devel	i386	4.1.2	i386-redhat-linux
libstdc++-devel	x86_64	4.1.2	x86_64-redhat-linux
libstdc++	i386	4.1.2	i386-redhat-linux
libstdc++	x86_64	4.1.2	x86_64-redhat-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
ncurses	i386	5.6	i386-redhat-linux-gnu

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

Name	Arch	Version	Platform
ncurses	i386	5.6	i386-redhat-linux-gnu

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.9

SUSE Linux Enterprise Server (SLES) 9

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the SUSE Linux Enterprise Server (SLES) 9 platform.

Release: SUSE LINUX Enterprise Server 9 (x86_64)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	2.05b	x86_64-suse-linux
binutils	x86_64	2.15.90.0.1.	x86_64-suse-linux
glibc-32bit	x86_64	9	x86_64-suse-linux
glibc-devel	x86_64	2.3.3	x86_64-suse-linux
glibc	x86_64	2.3.3	x86_64-suse-linux
glibc-profile ^a	x86_64	2.3.3	x86_64-suse-linux
grep	x86_64	2.5.1	x86_64-suse-linux
libgcc	x86_64	3.3.3	x86_64-suse-linux
libstdc++	x86_64	3.3.3	x86_64-suse-linux
make	x86_64	3.80	x86_64-suse-linux
python	x86_64	2.3.3	x86_64-suse-linux

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	3.3.3	x86_64-suse-linux
libstdc++-devel	x86_64	3.3.3	x86_64-suse-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

No additional requirements

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.10

SUSE Linux Enterprise 10

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the SUSE Linux Enterprise 10 platform.

Release: SUSE Linux Enterprise 10 (x86-64) OSS

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.1	x86_64-suse-linux
binutils	x86_64	2.16.91.0.5	x86_64-suse-linux
compat-libstdc++	x86_64	5.0.7	x86_64-suse-linux
glibc-32bit	x86_64	2.4	x86_64-suse-linux
glibc-devel	x86_64	2.4	x86_64-suse-linux
glibc	x86_64	2.4	x86_64-suse-linux
glibc-profile ^a	x86_64	2.4	x86_64-suse-linux
grep	x86_64	2.5.1a	x86_64-suse-linux
libgcc	x86_64	4.1.0	x86_64-suse-linux
libstdc++	x86_64	3.3.5	x86_64-suse-linux
make	x86_64	3.80	x86_64-suse-linux
python	x86_64	2.4.2	x86_64-suse-linux

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.1.0	x86_64-suse-linux
libstdc++-devel	x86_64	4.1.0	x86_64-suse-linux
libstdc++	x86_64	4.1.0	x86_64-suse-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

No additional requirements

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.11

SUSE Linux Professional 9.3

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the SUSE 9.3 platform.

Release: SUSE Linux Professional 9.3 (x86-64)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.0	x86_64-suse-linux
binutils	x86_64	2.15.94.0.2.	x86_64-suse-linux
glibc-32bit	x86_64	9.3	x86_64-suse-linux
glibc-devel	x86_64	2.3.4	x86_64-suse-linux
glibc	x86_64	2.3.4	x86_64-suse-linux
glibc-profile ^a	x86_64	2.3.4	x86_64-suse-linux
grep	x86_64	2.5.1a	x86_64-suse-linux
libgcc	x86_64	3.3.5	x86_64-suse-linux
libstdc++	x86_64	3.3.5	x86_64-suse-linux
make	x86_64	3.80	x86_64-suse-linux
python	x86_64	2.4	x86_64-suse-linux

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	3.3.5	x86_64-suse-linux
libstdc++-devel	x86_64	3.3.5	x86_64-suse-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

No additional requirements

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.12

SUSE Linux Professional 10.0

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the SUSE 10.0 platform.

Release: SUSE Linux Professional 10.0 (x86-64)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.0	x86_64-suse-linux
binutils	x86_64	2.16.91.0.2	x86_64-suse-linux
glibc-32bit	x86_64	2.3.5	x86_64-suse-linux
glibc-devel	x86_64	2.3.5	x86_64-suse-linux
glibc	x86_64	2.3.5	x86_64-suse-linux
glibc-profile ^a	x86_64	2.3.5	x86_64-suse-linux
grep	x86_64	2.5.1a	x86_64-suse-linux
libgcc	x86_64	4.0.2_20050901	x86_64-suse-linux
compat-libstdc++	x86_64	5.0.7	x86_64-suse-linux
make	x86_64	3.80	x86_64-suse-linux
python	x86_64	2.4.1	x86_64-suse-linux

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.0.2_20050901	x86_64-suse-linux
libstdc++-devel	x86_64	4.0.2_20050901	x86_64-suse-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

No additional requirements

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

D.1.13

SUSE Linux Professional 10.3

These are the required packages you need to run the compiled binaries from the PathScale compiler suite on the SUSE 10.3 platform.

Release: SUSE Linux Professional 10.3 (x86-64)

Compiler Base Packages:

```
pathscale-compilers-common-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-libs-lgpl-3.2-xxx_yyy_psc.i386.rpm  
pathscale-compilers-docs-3.2-xxx_yyy_psc.i386.rpm  
pathscale-sub-client-3.2-xxx_yyy_psc.i386.rpm
```

Name	Arch	Version	Platform
bash	x86_64	3.2	x86_64-suse-linux
binutils	x86_64	2.17.50.20070726	x86_64-suse-linux
glibc-32bit	x86_64	2.6.1	x86_64-suse-linux
glibc-devel	x86_64	2.6.1	x86_64-suse-linux
glibc	x86_64	2.6.1	x86_64-suse-linux
glibc-profile ^a	x86_64	2.6.1	x86_64-suse-linux
grep	x86_64	2.5.2	x86_64-suse-linux
libgcc	x86_64	4.2.1_20070724	x86_64-suse-linux
compat-libstdc++	x86_64	5.0.7	x86_64-suse-linux
make	x86_64	3.81	x86_64-suse-linux
python	x86_64	2.5.1	x86_64-suse-linux

Table Notes

^aOnly needed if using -profile to generate extra code

Package: pathscale-c-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-c++-3.2-xxx_yyy_psc.i386.rpm

Name	Arch	Version	Platform
gcc-c++	x86_64	4.2.1_20070724	x86_64-suse-linux
libstdc++-devel	x86_64	4.2.1_20070724	x86_64-suse-linux

Package: pathscale-f90-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-pathdb-3.2-xxx_yyy_psc.x86_64.rpm

No additional requirements

Package: pathscale-pathopt2-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Package: pathscale-sub-server-3.2-xxx_yyy_psc.i386.rpm

No additional requirements

Index

C

CD, description of 3-3
Compiler test program 3-32

D

Distributions, supported 2-1
Documentation, list of 1-2

E

Environment variables 3-19, 3-20, 3-22, 3-24, 3-26, 3-28, 3-29, 5-4, A-5

F

File formats, download 2-2

I

Installation
 change default directory 3-10
 multiple compiler versions 3-11
 non-root tar file 3-16
 using RPMs 3-9, 3-13
ISO image, description of 3-3

M

MAC address format 3-5
man pages, accessing 3-20
Mount point for ISO image 3-8

N

Network, setting compiler environment for 3-21
Non-root tar file, description of 3-2

P

PSC_WAIT4COMPILER A-5

R

RPM installation 3-9

RPMs

base set of 4-1

individual compilers and programs 4-1

S

Seat, definition of 3-25

Subscription file

editing 3-27

ordering 2-2, 3-4, 3-5

Subscription types 3-3

T

tar file, description of 3-2

Troubleshooting

installation B-1–B-2

subscriptions B-3–B-11

U

Uninstalling

RPMs 6-1

Upgrading 3-7

