

IBM XL C/C++ Advanced Edition V7.0 for Linux[®]



Installation Guide

Version 7.0

IBM XL C/C++ Advanced Edition V7.0 for Linux[®]



Installation Guide

Version 7.0

Note!

Before using this information and the product it supports, read the information in “Notices” on page 19.

First Edition (June 2004)

This edition applies to IBM XL C/C++ Advanced Edition V7.0 for Linux and to all subsequent releases and modifications until otherwise indicated in new editions.

IBM welcomes your comments. You can send them to compinfo@ca.ibm.com. Be sure to include your e-mail address if you want a reply. Include the title and order number of this book, and the page number or topic related to your comment.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1997, 2004. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Installation guide. 1

XL C/C++ packaging and filesets	1
Installing IBM XL C/C++ Advanced Edition V7.0 for Linux.	2
System prerequisites.	2
Prerequisite tasks or conditions	2
Installation	3
Product migration installation for RHEL3.	6
Querying for installed packages	8
Enabling the XL C/C++ man pages.	9
Setting the correct NLSPATH.	9
Configuring the compiler.	11
Viewing the documentation	12

Setting up the environment for the invocation commands.	13
Testing the installation.	13
Uninstalling XL C/C++	14

Troubleshooting. 15

Troubleshooting for RHEL3	15
Troubleshooting for SLES9	16
Troubleshooting for both RHEL3 and SLES9	16

Notices 19

Trademarks and service marks	20
--	----

Installation guide

This document contains essential information about installing IBM XL C/C++ Advanced Edition V7.0 for Linux. Please read it carefully before installing IBM XL C/C++ Advanced Edition V7.0 for Linux. Please also read the **README** file on the CD-ROM, which contains the most current information about IBM XL C/C++ Advanced Edition V7.0 for Linux. After you install the product, you can find the **README** file in the *compiler-path/vacpp/7.0* directory.

XL C/C++ packaging and filesets

The following packages are supplied on the CD. Use rpm to review them:

Required packages

Package Name	package Description
xlsmp.msg.rte	SMP run-time messages
xlsmp.rte	SMP run-time dynamic libraries
xlsmp.lib	SMP run-time static libraries
xlmass.lib	IBM Mathematical Acceleration Subsystem (MASS) component
vac.lic	XL C/C++ license
vacpp.rte	C++ run-time environment
vacpp.rte.lnk	C++ run-time environment links
vac.lib	C compiler libraries
vac.cmp	C compiler
vacpp.lib	C++ compiler libraries
vacpp.cmp	C++ Compiler
vacpp.samples	XL C/C++ samples
vacpp.help	XL C/C++ documentation

The supported locales are as follows:

- en_US
- en_US.utf8
- ja_JP
- ja_JP.eucjp
- ja_JP.utf8
- zh_CN
- zh_CN.gb18030
- zh_CN.gb2312
- zh_CN.gbk
- zh_CN.utf8

Installing IBM XL C/C++ Advanced Edition V7.0 for Linux

The IBM XL C/C++ Advanced Edition V7.0 for Linux installation uses the Red Hat Package Manager (RPM), which is the standard tool for installing Linux packages. RPM is shipped with the operating system.

The high-level steps are:

1. Become the root user, or a user with administrator privileges.
2. Install the packages.
3. Enable the compiler man pages.
4. Set the correct NLSPATH.
5. Configure the compiler.
6. Set up the environment for the invocation commands.
7. Test the installation.

System prerequisites

- **Operating system:** A supported Linux distribution:
 - Red Hat Enterprise Linux AS for POWER V3 (RHEL 3), Update 3
 - SUSE LINUX Enterprise Server 9 for POWER (SLES 9)
- **Hardware:** An IBM pSeries™, iSeries™ or BladeCenter™ computer that is supported by the Linux distribution.
- **Required hard disk space:**
 - Approximately 100 MB for product packages
 - High levels of optimization, may require significant amounts of additional paging and temporary disk space.
- **Required software:**
 - HTML browser (to access documentation)
 - PDF viewer (to access documentation)

Prerequisite tasks or conditions

You must have root user access to install XL C/C++.

Checking for required packages

You must install the following packages before you install the compiler packages:

RHEL3	
GCC prerequisites	Version Requirements
gcc	3.2.3
gcc-c++	3.2.3
glibc-devel	2.3.2
libstdc++-devel	3.2.3

SLES9	
GCC prerequisites	Version Requirements
gcc	3.3.3
gcc-c++	3.3.3
gcc-64bit	9
glibc-devel-64bit	9
libstdc++-devel-64bit	9

SLES9	
java2	1.3.1
java2-jre	1.3.1

Note: For installing on RHEL3, both 32-bit and 64-bit glibc-devel and libstdc++-devel packages are required. To make sure you have these packages installed before you install the compiler, follow the instruction in “Troubleshooting for RHEL3” on page 15. You cannot use the following command to check whether these packages are installed, because the names of the RPM packages on RHEL3 are structured in a way that they have the same name regardless whether they are 32-bit or 64-bit RPM package. As a result, the output does not indicate whether 32-bit, 64-bit or both RPM packages are installed.

Use the following command to determine if a package is installed:

```
rpm -qa | grep package_name
```

For example, to see whether `vac.cmp` is installed, query `vac.cmp` package as follows:

```
rpm -qa | grep vac.cmp
```

If `vac.cmp` version 7.0 is installed, you will get the following output as the result of query:

```
vac.cmp-7.0.0-0
```

Checking for other packages

The following items require other components or are themselves prerequisites for other components. The **vac_configure** utility is required because it must be run after the initial installation in order for the compiler to work. The KDE desktop environment and viewers for accessing the compiler documentation are optional.

Other package dependencies

Optional?	Component	Prerequisite RPM package(s)	Query
Required	vac_configure utility	perl	<code>rpm -qf /usr/bin/perl</code>
Optional	Mozilla or Konqueror browser	RPM packages for the KDE environment	<code>rpm -q mozilla</code> <code>rpm -q kdebase3-konqueror</code>
Optional	PDF document viewer	RPM packages for the KDE environment	<code>rpm -q xpdf</code>

Verifying space requirements

XL C/C++ packages require about 100 MB of hard disk space. This amount includes the optional samples and documentation. Compiling at higher levels of optimization may require large amounts of temporary disk space.

Use the following command to determine the amount of space available for the compiler installation in the default installation location:

```
df -h /opt
```

Installation

By default, the compiler installs in the `/opt/ibmcmp` directory.

Installation of all the packages using a single command

If your current working directory contains all of the RPM packages for XL C/C++ and no other RPM packages, you can install XL C/C++ using one single command as follows:

```
rpm -ivh *.rpm (default installation)
```

```
rpm -ivh *.rpm --prefix=installation-path (non-default installation)
```

The following table lists the packages and their prerequisites for both RHEL and SLES9.

XL C/C++ packages and their prerequisites

Package name	Prerequisites	Description	Relocation
xlsmp.msg.rte	none	SMP run-time messages	must be installed in the same location
xlsmp.rte	xlsmp.msg.rte	SMP run-time dynamic libraries	
xlsmp.lib	xlsmp.msg.rte xlsmp.rte	SMP run-time static libraries	
xlmass.lib	none	IBM Mathematical Acceleration Subsystem (MASS) component	any location
vac.lic	none	XL C/C++ license	any location
vacpp.rte	none	C++ run-time environment	vacpp.rte and vacpp.rte.lnk must be installed in the same location
vacpp.rte.lnk	vacpp.rte	C++ run-time environment links	
vac.cmp	vac.lib vac.lic xlmass.lib xlsmp.lib xlsmp.msg.rte xlsmp.rte	C compiler	vac.cmp, vac.lib, vacpp.cmp and vacpp.lib must be installed in the same location
vac.lib	none	C compiler libraries	
vacpp.cmp	vac.cmp vac.lic vac.lib vacpp.lib vacpp.rte vacpp.rte.lnk xlmass.lib xlsmp.lib xlsmp.msg.rte xlsmp.rte	C++ compiler	
vacpp.lib	vac.lib	C++ compiler libraries	
vacpp.samples	none	Example programs	any location (optional)
vacpp.help	none	Compiler documentation in HTML and PDF formats	any location (optional)

Default installation on RHEL3

Issue the following commands to install XL C/C++ to the default location, **/opt/ibmcmp**. To avoid dependency errors during installation, install the compiler packages in the following order:

```

rpm -ivh xlsmp.msg.rte-1.5.0-0.ppc64pseries.rpm
rpm -ivh xlsmp.rte-1.5.0-0.ppc64pseries.rpm
rpm -ivh xlsmp.lib-1.5.0-0.ppc64pseries.rpm
rpm -ivh xlmass.lib-4.1.0-0.ppc64pseries.rpm
rpm -ivh vac.lic-7.0.0-0.ppc64pseries.rpm
rpm -ivh vac.lib-7.0.0-0.ppc64pseries.rpm
rpm -ivh vac.cmp-7.0.0-0.ppc64pseries.rpm
rpm -ivh vacpp.rte-7.0.0-0.ppc64pseries.rpm
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64pseries.rpm
rpm -ivh vacpp.lib-7.0.0-0.ppc64pseries.rpm
rpm -ivh vacpp.cmp-7.0.0-0.ppc64pseries.rpm

```

The sample programs and product documentation packages have no dependency on other RPM packages and can be installed in any order using the following commands:

```

rpm -ivh vacpp.help-7.0.0-0.ppc64pseries.rpm
rpm -ivh vacpp.samples-7.0.0-0.ppc64pseries.rpm

```

Note: If you have already installed the XL Fortran compiler, `xlsmp.msg.rte`, `xlsmp.rte`, `xlsmp.lib` and `xlmass.lib` will already be installed. RPM will prevent you from installing these components a second time.

Non-default installation on RHEL3

Issue the following commands to install XL C/C++ to a non-default location. To avoid dependency errors during installation, install the compiler packages in the following order:

```

rpm -ivh xlsmp.msg.rte-1.5.0-0.ppc64pseries.rpm --prefix=smprt-path
rpm -ivh xlsmp.rte-1.5.0-0.ppc64pseries.rpm --prefix=smprt-path
rpm -ivh xlsmp.lib-1.5.0-0.ppc64pseries.rpm --prefix=smprt-path
rpm -ivh xlmass.lib-4.1.0-0.ppc64pseries.rpm --prefix=xlmass-path
rpm -ivh vac.lic-7.0.0-0.ppc64pseries.rpm --prefix=vaclic-path
rpm -ivh vac.lib-7.0.0-0.ppc64pseries.rpm --prefix=compiler-path
rpm -ivh vac.cmp-7.0.0-0.ppc64pseries.rpm --prefix=compiler-path
rpm -ivh vacpp.rte-7.0.0-0.ppc64pseries.rpm --prefix=vacpprt-path
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64pseries.rpm --prefix=vacpprt-path
rpm -ivh vacpp.lib-7.0.0-0.ppc64pseries.rpm --prefix=compiler-path
rpm -ivh vacpp.cmp-7.0.0-0.ppc64pseries.rpm --prefix=compiler-path

```

For example:

```

rpm -ivh xlsmp.msg.rte-1.5.0-0.ppc64pseries.rpm --prefix=/common
rpm -ivh xlsmp.rte-1.5.0-0.ppc64pseries.rpm --prefix=/common
rpm -ivh xlsmp.lib-1.5.0-0.ppc64pseries.rpm --prefix=/common
rpm -ivh xlmass.lib-4.1.0-0.ppc64pseries.rpm --prefix=/common_xlmass
rpm -ivh vac.lic-7.0.0-0.ppc64pseries.rpm --prefix=/license
rpm -ivh vac.lib-7.0.0-0.ppc64pseries.rpm --prefix=/compiler
rpm -ivh vac.cmp-7.0.0-0.ppc64pseries.rpm --prefix=/compiler
rpm -ivh vacpp.rte-7.0.0-0.ppc64pseries.rpm --prefix=/runtime
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64pseries.rpm --prefix=/runtime
rpm -ivh vacpp.lib-7.0.0-0.ppc64pseries.rpm --prefix=/compiler
rpm -ivh vacpp.cmp-7.0.0-0.ppc64pseries.rpm --prefix=/compiler

```

Default installation on SLES9

Issue the following commands to install XL C/C++ to the default location, `/opt/ibmcmp`. To avoid dependency errors during installation, install the compiler packages in the following order:

```

rpm -ivh xlsmp.msg.rte-1.5.0-0.ppc64.rpm
rpm -ivh xlsmp.rte-1.5.0-0.ppc64.rpm
rpm -ivh xlsmp.lib-1.5.0-0.ppc64.rpm
rpm -ivh xlmass.lib-4.1.0-0.ppc64.rpm
rpm -ivh vac.lic-7.0.0-0.ppc64.rpm
rpm -ivh vac.lib-7.0.0-0.ppc64.rpm
rpm -ivh vac.cmp-7.0.0-0.ppc64.rpm

```

```
rpm -ivh vacpp.rte-7.0.0-0.ppc64.rpm
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64.rpm
rpm -ivh vacpp.lib-7.0.0-0.ppc64.rpm
rpm -ivh vacpp.cmp-7.0.0-0.ppc64.rpm
```

The sample programs and product documentation packages have no dependency on other RPM packages and can be installed in any order.

```
rpm -ivh vacpp.help-7.0.0-0.ppc64.rpm
rpm -ivh vacpp.samples-7.0.0-0.ppc64.rpm
```

Note: If you have already installed the XL Fortran compiler, **xlsmprte**, **xlsmplib** will already be installed. RPM will prevent you from installing these components a second time.

Non-default installation on SLES9

Issue the following commands to install XL C/C++ to a non-default location. To avoid dependency errors during installation, install the compiler packages in the following order:

```
rpm -ivh xlsmprte.msg.rte-1.5.0-0.ppc64.rpm --prefix=smprt-path
rpm -ivh xlsmprte.rte-1.5.0-0.ppc64.rpm --prefix=smprt-path
rpm -ivh xlsmplib.lib-1.5.0-0.ppc64.rpm --prefix=smprt-path
rpm -ivh xlmass.lib-4.1.0-0.ppc64.rpm --prefix=xlmass-path
rpm -ivh vaclic-7.0.0-0.ppc64.rpm --prefix=vaclic-path
rpm -ivh vaclib-7.0.0-0.ppc64.rpm --prefix=compiler-path
rpm -ivh vaccmp-7.0.0-0.ppc64.rpm --prefix=compiler-path
rpm -ivh vacpp.rte-7.0.0-0.ppc64.rpm --prefix=vacpprt-path
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64.rpm --prefix=vacpprt-path
rpm -ivh vacpp.lib-7.0.0-0.ppc64.rpm --prefix=compiler-path
rpm -ivh vacpp.cmp-7.0.0-0.ppc64.rpm --prefix=compiler-path
```

For example:

```
rpm -ivh xlsmprte.msg.rte-1.5.0-0.ppc64.rpm --prefix=/common
rpm -ivh xlsmprte.rte-1.5.0-0.ppc64.rpm --prefix=/common
rpm -ivh xlsmplib.lib-1.5.0-0.ppc64.rpm --prefix=/common
rpm -ivh xlmass.lib-4.1.0-0.ppc64.rpm --prefix=common_xlmass
rpm -ivh vaclic-7.0.0-0.ppc64.rpm --prefix=/license
rpm -ivh vaclib-7.0.0-0.ppc64.rpm --prefix=/compiler
rpm -ivh vaccmp-7.0.0-0.ppc64.rpm --prefix=/compiler
rpm -ivh vacpp.rte-7.0.0-0.ppc64.rpm --prefix=/runtime
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64.rpm --prefix=/runtime
rpm -ivh vacpp.lib-7.0.0-0.ppc64.rpm --prefix=/compiler
rpm -ivh vacpp.cmp-7.0.0-0.ppc64.rpm --prefix=/compiler
```

Product migration installation for RHEL3

Note: As XL C/C++ version 7.0 is the first version supported on SLES9, product migration installation does not apply to SLES9.

Product migration installation

If you have the previous version of the compiler installed, it is recommended that you uninstall any previous versions of the compiler prior to installation. However, if you want to have more than one version of the compiler on the system, you can follow one of the methods described in this section.

Method 1: Installing to a different location: You can install version 7.0 to a different location from version 6.0.

For example, if you installed version 6.0 compiler to the default location, **/opt/ibmcmp**, follow the instructions in “Non-default installation on RHEL3” on page 5 to install version 7.0 packages to a different location.

If you did not use the default installation for version 6.0 and installed it to a non-default location, you can install version 7.0 compiler to the default location. Follow the instructions in “Default installation on RHEL3” on page 4. You can also install version 7.0 to a non-default location by following the instructions in “Non-default installation on RHEL3” on page 5.

Method 2: Installing the compiler to the same location: You can also install version 7.0 packages to the same location that version 6.0 packages reside in. In this case, you will only have version 7.0 run-time packages installed on that location and the version 6.0 compiler will use version 7.0 run-time shared libraries. To avoid future dependency errors, you must remove the following package before you install version 7.0 packages to the same location.

```
rpm -e xlsmp.msg.rte-1.3.7-2 --nodeps
```

Assuming all the packages for version 6.0 are installed, use the following commands to install version 7.0 packages to the same location. To avoid dependency errors during installation, install the compiler packages in the following order:

```
rpm -ivh xlsmp.msg.rte-1.5.0-0.ppc64pseries.rpm --prefix <V1.3_smprt_path>
rpm -ivh xlsmp.rte-1.5.0-0.ppc64pseries.rpm --prefix <V1.3_smprt_path> --force
rpm -ivh xlsmp.lib-1.5.0-0.ppc64pseries.rpm --prefix <V1.3_smprt_path>
rpm -ivh xlmass.lib-4.1.0-0.ppc64pseries.rpm --prefix <xlmass_path>
rpm -ivh vac.lic-7.0.0-0.ppc64pseries.rpm --prefix <V6.0_license_path>
rpm -ivh vac.lib-7.0.0-0.ppc64pseries.rpm --prefix <V6.0_compiler_path>
rpm -ivh vac.cmp-7.0.0-0.ppc64pseries.rpm --prefix <V6.0_compiler_path>
rpm -ivh vacpp.rte-7.0.0-0.ppc64pseries.rpm --prefix <V6.0_runtime_path> --force
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64pseries.rpm --prefix <V6.0_runtime_path>
rpm -ivh vacpp.lib-7.0.0-0.ppc64pseries.rpm --prefix <V6.0_compiler_path>
rpm -ivh vacpp.cmp-7.0.0-0.ppc64pseries.rpm --prefix <V6.0_compiler_path>
```

where

<V1.3_smprt_path> is where previous version 1.3 of smprt was installed.

<V6.0_compiler_path> is where previous version 6.0 of the compiler was installed.

<V6.0_license_path> is where previous version 6.0 of the compiler license was installed.

<V6.0_runtime_path> is where previous version 6.0 of the compiler run-time was installed.

<xlmass_path> can be any location on the system.

Notes:

1. The **--force** option must be used on the run-time packages to avoid file conflicts. If you do not use the **--force** option you will get warnings such as the following:
file /opt/ibmcmp/lib/libibmc++.so.1 from install of vacpp.rte-7.0.0-0 conflicts with file from package vacpp.rte-6.0.1-0
2. If you use method 2 for migration installation, you must use **vac_configure** manually. Do not use **new_install** when more than one version of the compiler is installed on the system.

3. When you use method 2, version 7.0 run-time packages will be used by both versions of the compiler. If you need version 6.0 run-time packages to run with version 6.0 run-time packages, you must use Method 1.

PTF updates

With the PTF (Program Temporary Fix) updates, all of the packages except `vac.lic` are shipped.

It is recommended that you uninstall any previous version of the compiler prior to applying PTF updates. In this case, you should use the `vac.lic` you have from your previous version. However, you have the option to keep the previous version of the compiler or uninstall it based on your preferences when applying PTF updates.

Method 1: keeping the previous version of the compiler and applying the PTF updates: If you want to keep the previous version of the compiler on your system, you must install the PTF to a different location. In this case, there will be no need to reinstall `vac.lic`, but note that if you do not uninstall the previous version of the compiler, you must use `vac_configure` option. Do not use `new_install`.

Later, once you have verified that the PTF is what you want and decide to uninstall the previous version, you can uninstall everything except `vac.lic`. If you do not uninstall the previous version of the compiler from the system, and attempt to apply the PTF in the same install location, the installation fails, because the packages are installed in a common location, which is not specific to the version of the product. You can follow these steps to apply the PTF:

1. Complete the instructions with the new RPM packages as mentioned in the "Installation" on page 3. Since the `vac.lic` package is already installed, you do not need to install it again.
2. Configure the compiler using `vac_configure`. Do not use the `new_install` utility.
3. If you decided to keep the previous version of the compiler when applying the PTF updates and later, you want to uninstall the old compiler, follow the instructions in "Uninstalling XL C/C++" on page 14. Do not uninstall `vac.lic`. If you try to uninstall this package, you will receive an error message because this package is a prerequisite for other packages that are installed on your system.

Method 2: uninstalling the previous version of the compiler and applying the PTF updates: If you do not want to have the previous version of the compiler when you apply the PTF updates, follow the instructions described in the "Uninstalling XL C/C++" on page 14 and apply the PTF updates using the instruction in "Installation" on page 3. You should use the `vac.lic` that you have from the previous version.

Querying for installed packages

To query an individual package, issue a command such as the following:

```
rpm -q vac.cmp
```

If the installation is not successful, you will get a message indicating that the package has not been installed. If the package has been installed properly, the result should be:

```
vac.cmp-V.R.M-F
```

Where *V.R.M-F* is the Version.Release.Modification.Fix level of the compiler installed on the system.

To query all packages and search for *vac*, issue the following command:

```
rpm -qa | grep vac
```

If the package is not installed, the output of the command will be null.

Enabling the XL C/C++ man pages

Man pages are provided for the compiler invocation commands and other utilities that are provided with the compiler.

Before you can read the man pages, you must add the XL C/C++ man directory to the MANPATH environment variable as follows:

1. Open a terminal window.
2. Enter the following command:

```
export LANG=locale
```

The supported locales for manual pages are as follows:

- en_US
- en_US.utf8
- ja_JP
- ja_JP.eucjp

For example, to view man pages in English, enter the following command:

```
export LANG=en_US
```

3. Add **/opt/ibmcmp/vacpp/7.0/man** to the beginning of the MANPATH environment variable, as shown in the following table.

How to update the MANPATH environment variable

Shell	Command
bash	export MANPATH=/opt/ibmcmp/vacpp/7.0/man:\$MANPATH

To invoke a man page, enter **man** followed by the command; for example:

```
man xlc.
```

To leave the man page, type: **q**.

Setting the correct NLSPATH

After installation, you must set the NLSPATH environment variable so that the run-time environment finds the appropriate message catalogs.

Issue the following command:

```
export NLSPATH=$NLSPATH:  
smprrt-path/msg/%L/%N:  
vacpprrt-path/msg/%L/%N:  
compiler-path/vacpp/7.0/msg/%L/%N
```

where:

- *smprrt-path* is the installation location of the SMP run-time
- *vacpprrt-path* is the installation location of the C++ run-time environment
- *compiler-path* is the installation location of the C and C++ compilers

Note: If the default installation location is used, *smprt-path*, *vacpprt-path*, and *compiler-path* will all be */opt/ibmcmp*.

Configuring the compiler

In order to configure the compiler, you should have root or administrator privileges if you run **new_install**. However, if you configure the compiler using **vac_configure** and your output configuration file, *vac.cfg*, is written to a location where you have write permission, you do not need root or administrative privileges.

You can use **new_install** to generate the default configuration file automatically. The **new_install** command queries the RPM database to search for the paths to the XL C/C++ packages and also search for the path to the 32-bit GCC (*gcc32path*) and the 64-bit GCC (*gcc64path*) in the PATH environment variable and generates the */etc/opt/ibmcmp/vac/7.0/vac.cfg* configuration file. It uses the values obtained to execute **vac_configure**. When running **new_install** on SLES9 platform, you will be required to accept the license agreement before the configuration file is generated. This step is not required on the RHEL3.

1. Change to the directory that contains the **new_install** and **vac_configure** executables:

```
cd /opt/ibmcmp/vacpp/7.0/bin
```

or, if you installed XL C/C++ in a non-default location:

```
cd compiler-path/vacpp/7.0/bin
```

2. Run the following command:

```
./new_install
```

The **new_install** command executes the following command:

```
vac_configure
-gcc gcc32path
-gcc64 gcc64path
-install
-mass xlmass-path/xlmass/4.1
-smprt smprt-path/xlsmpt/1.5
-vaclic vaclic-path/vac/7.0
-vacpprt vacpprt-path/vacpp/7.0
-vac compiler-path/vac/7.0
-vacpp compiler-path/vacpp/7.0
compiler-path/vac/7.0/etc/vac.base.cfg
```

If the default configuration file exists before you invoke **new_install**, **new_install** will back up the existing configuration file. The name of the backup file will be displayed to the screen.

Note: You should use **vac_configure** manually if multiple instances of XL C/C++ or GCC are installed on your system.

If **new_install** exits with an error, you will need to run the **vac_configure** utility manually, as described in the “The *vac_configure* utility” on page 11.

The following table describes some of the attributes in the generated configuration file.

Linux-specific configuration attributes

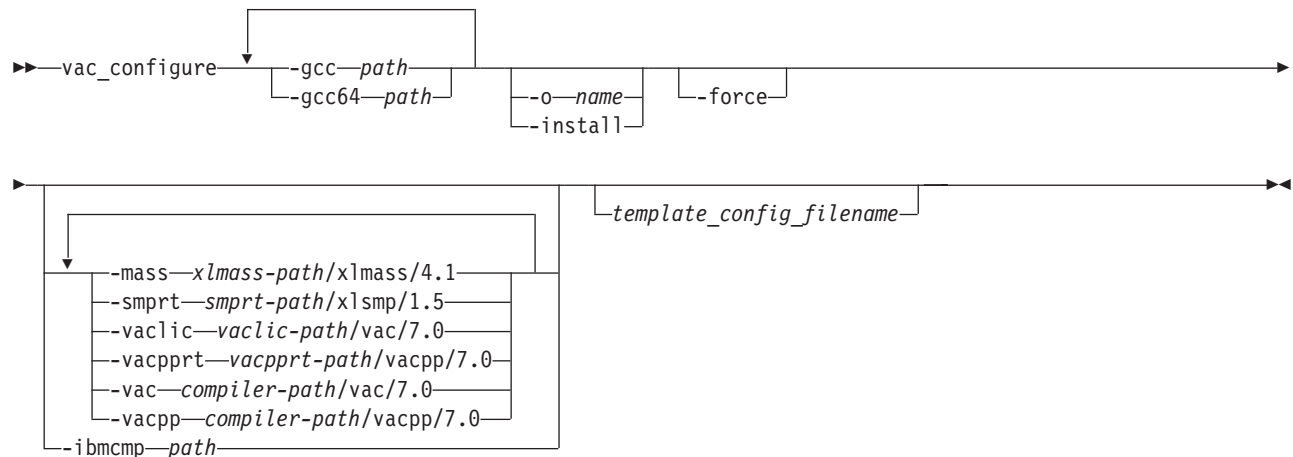
Attribute	Contents	Additional information
gcc_path gcc_path_64	The location of the GCC executable, in 32-bit mode or 64-bit mode	The gcc command must be located in the bin directory under the specified path.
gcc_libs gcc_libs_64	A comma-separated list of GCC libraries, in 32-bit mode or 64-bit mode	The gcc returns this list to the vac_configure utility.
gcc_libdirs gcc_libdirs_64	A comma-separated list of directories that contain GCC libraries, in 32-bit mode or 64-bit mode	The gcc returns this list to the vac_configure utility.

The vac_configure utility

The **vac_configure** utility is automatically invoked by the **new_install** command. You can invoke the **vac_configure** utility directly in the following situations:

- You receive an error from the **new_install** command.
- You successfully run **new_install** but later want to change the default configuration file, for example if you install a new version of the GCC.
- You want to create additional configuration files.

The **vac_configure** command has the following syntax:



where:

-gcc path

Specifies the path where the GCC bin directory is installed. In the configuration file, the *gcc_path* attribute is set equal to this path.

For example, if the GCC command is **/usr/bin/gcc**, you would specify

-gcc /usr

-gcc64 path

Specifies the path where the 64-bit GCC bin directory is installed. In the configuration file, the *gcc_path_64* attribute is set equal to this path.

-o name

Specifies the name of the configuration file to generate. By default, output is written to the display.

- install** Generates a file with the required name and the required location for the default configuration file: `/etc/opt/ibmcmp/vac/7.0/vac.cfg`
- force** Forces the **vac_configure** utility to overwrite any existing output file with the specified name and path. By default, if you don't use *force*, **vac_configure** issues an error message and stops if the specified file already exists.
- mass** *xlmass-path/xlmass/4.1*
Specifies the path for the **xlmass.lib** package. By default the path is `/opt/ibmcmp/xlmass/4.1`.
- smprt** *smprt-path/xlsmp/1.5*
Specifies the path for the **xlsmp.msg.rte**, **xlsmp.rte**, and **xlsmp.lib** packages. By default, the path is `/opt/ibmcmp/xlsmp/1.5`.
- vaclic** *vaclic-path/vac/7.0*
Specifies the path for the **vac.lic** package. By default, this is `/opt/ibmcmp/vac/7.0`.
- vacpprt** *vacpprt-path/vacpp/7.0*
Specifies the path for the **vacpp.rte** and **vacpp.rte.lnk** package. By default, this is `/opt/ibmcmp/vacpp/7.0`.
- vacpp** *compiler-path/vacpp/7.0*
Specifies the path for the **vacpp.cmp** package. By default, this is `/opt/ibmcmp/vacpp/7.0`.
- vac** *compiler-path/vac/7.0*
Specifies the path for the **vac.cmp** package. By default, this is `/opt/ibmcmp/vac/7.0`.
- ibmcmp** *path*
Alternatively specifies the path where all of the XL C/C++ packages (**xlsmp.msg.rte**, **xlsmp.rte**, **xlsmp.lib**, **vac.lic**, **vacpp.rte**, **vacpp.rte.lnk**, **vac.cmp**, and **vacpp.cmp**) are installed. In this case, *path* is the same as *smprt-path*, *vaclic-path*, *runtime-path*, and *compiler-path*. You can not specify **ibmcmp** and also specify the individual paths.
- template_config_filename**
The input file that is used to construct the configuration file. By default, this is `/opt/ibmcmp/vac/7.0/etc/vac.base.cfg`. If you relocated the **vac.cmp** package but want to use the default template, specify:
`compiler-path/vac/7.0/etc/vac.base.cfg`.

Viewing the documentation

The following documentation is provided with XL C/C++:

- README file** A README file is located in the root directory of the installation CD. This file is installed in the `/opt/ibmcmp/vacpp/7.0` directory.
- PDF books** The PDF versions of the XL C/C++ documentation are stored in the `/doc/$LANG/pdf` directory of the installation CD. When you install the **vacpp.help** package, the PDF files are copied to the `/opt/ibmcmp/vacpp/7.0/doc/$LANG/pdf` directory.
- HTML files** When you install the **vacpp.help** package, an HTML version of the XL C/C++ documentation is installed in the

`/opt/ibmcmp/vacpp/7.0/doc/$LANG/html` directory. To view the HTML files, open the file **index.htm** in a browser such as Mozilla or Konqueror.

Man pages Man pages are provided for the compiler invocation commands (such as `xlc`) and the following additional commands: **`vac_configure`**, **`new_install`**, **`gxlc`**, **`gxlc++`**, **`showpdf`**, **`mergepdf`**, **`resetpdf`**, and **`cleanpdf`**. The man pages are installed in the `/opt/ibmcmp/vacpp/7.0/man/$LANG/man1` directory.

Setting up the environment for the invocation commands

XL C/C++ is not automatically installed in `/usr/bin`. To invoke the compiler without having to specify the full path, do one of the following steps:

- Create symbolic links for the specific drivers contained in `/opt/ibmcmp/vac/7.0/bin` and `/opt/ibmcmp/vacpp/7.0/bin` to `/usr/bin` as follows:

```
ln -s <realpath> <targetpath>
```

For example:

```
ln -s /opt/ibmcmp/vacpp/7.0/bin/xlc /usr/bin/xlc
```

- Add `/opt/ibmcmp/vac/7.0/bin` and `/opt/ibmcmp/vacpp/7.0/bin` to the PATH environment variable.

Testing the installation

To test the product install and the critical search paths, try building the following simple C and C++ programs.

1. Create the following C program and name the source file `hello.c`:

```
#include <stdio.h>
int main(void)
{
    printf("Hello World!\n");
    return 0;
}
```

2. Use the `xlc` command to compile the test program. For example:

```
/opt/ibmcmp/vac/7.0/bin/xlc hello.c -o hello
```

3. Run the program:

```
./hello
```

The expected result is that "Hello World!" is displayed on the screen.

4. Check the exit code of the program:

```
echo $?
```

The result should be zero.

5. Create the following C++ program and name the source file `hello.c++`:

```
#include <iostream>
int main()
{
    std::cout << "Hello World!" << std::endl;
    return 0;
}
```

6. Use the `xlc++` command to compile the test program. For example:

```
/opt/ibmcmp/vacpp/7.0/bin/xlc++ hello.c++ -o hello
```

7. Run the program:

```
./hello
```

The expected result is that "Hello World!" is displayed on the screen.

8. Check the exit code of the program:

```
echo $?
```

The result should be zero.

Uninstalling XL C/C++

You must have root user access to uninstall this product. Note that the order of uninstalling packages is the reverse of that in which they were installed (LIFO). Uninstall the compiler packages in the following order to avoid dependency errors during uninstallation (both default and non-default install location use the same set of commands) :

- **rpm -e vacpp.cmp-7.0.0-0**
- **rpm -e vacpp.lib-7.0.0-0**
- **rpm -e vacpp.rte.lnk-7.0.0-0**
- **rpm -e vacpp.rte-7.0.0-0**
- **rpm -e vac.cmp-7.0.0-0**
- **rpm -e vac.lic-7.0.0-0**
- **rpm -e vac.lib-7.0.0-0**
- **rpm -e xlmass.lib-4.1.0-0**
- **rpm -e xlsmp.lib-1.5.0-0**
- **rpm -e xlsmp.rte-1.5.0-0**
- **rpm -e xlsmp.msg.rte-1.5.0-0**

Notes:

1. Some packages may not uninstall if they are required by other packages. For example, the SMP runtime (xlsmp.rte) might be a shared component if IBM XL Fortran is also installed on the same system.
2. The above commands will not remove any vac.cfg that were generated by **new_install** or **vac_configure**.

The sample programs and product documentation do not have any package dependencies. To uninstall them, you can issue the following commands in any order.

```
rpm -e vacpp.samples-7.0.0-0  
rpm -e vacpp.help-7.0.0-0
```

Troubleshooting

This section describes possible problems you may encounter when you install IBM XL C/C++ Advanced Edition V7.0 for Linux.

Troubleshooting for RHEL3

You must have both ppc and ppc64 glibc-devel and libstdc++-devel packages installed to successfully install and configure the product. If you only have the ppc RPM packages installed but not the ppc64 RPM packages when you run **new_install** or **vac_configure**, you will receive one of the following error messages:

```
Could not determine location of 64-bit gcc.  
Suggestion: 64-bit glibc-devel, 64-bit libstdc++-devel might need to be  
installed  
  
Could not determine location of 32-bit gcc.  
Suggestion: 32-bit glibc-devel, 32-bit libstdc++-devel might need to be  
installed
```

In this case, you should check whether both of the 64-bit and 32-bit RPM packages of glibc-devel and libstdc++-devel are installed on the system. On RHEL3 you cannot tell whether the 32-bit or the 64-bit version of GCC is installed by querying the packages because both the 32-bit and 64-bit packages have the exact same RPM name. You can, however, check this by compiling a test case in 32-bit and 64-bit mode.

For example, to check whether the 32-bit or the 64-bit version of GCC is installed you can create two test cases helloWorld.c and helloWorld.cpp and try to compile the files as follows:

For 32-bit mode:

```
gcc helloWorld.c  
gcc helloWorld.cpp
```

If the programs compile successfully without any error message, it indicates that you have 32-bit RPM packages installed. Otherwise if you get an error message, it means that you do not have the 32-bit RPM packages installed and you need to install the necessary packages, using the following commands:

```
rpm -ivh glibc-devel-V.R.M-F.ppc.rpm  
rpm -ivh libstdc++-devel-V.R.M-F.ppc.rpm
```

Where *V.R.M-F* is the Version.Release.Modification.Fix level of the package installed on the system.

For example, you can use the following commands:

```
rpm -ivh glibc-devel-2.3.2-95.3.ppc.rpm  
rpm -ivh libstdc++-devel-3.2.3-20.ppc.rpm
```

For 64-bit mode:

```
gcc -m64 helloWorld.c  
gcc -m64 helloWorld.cpp
```

If the programs compile successfully without any error message, it indicates that you have 64-bit RPM packages installed. Otherwise if you get an error message, it means that you do not have the 64-bit RPM packages installed and you need to install the necessary packages, using the following commands:

```
rpm -ivh glibc-devel-V.R.M-F.ppc64.rpm
rpm -ivh libstdc++-devel-V.R.M-F.ppc64.rpm
```

Where *V.R.M-F* is the Version.Release.Modification.Fix level of the package installed on the system.

For example, you can use the following commands:

```
rpm -ivh glibc-devel-2.3.2-95.3.ppc64.rpm
rpm -ivh libstdc++-devel-3.2.3-20.ppc64.rpm
```

After you install the packages, run **new_install** again and the product should install without any error.

Troubleshooting for SLES9

Java2 packages are required to run **new_install**. If you do not have java2 or java2-jre installed on your machine and try to run **new_install**, you will get the following error message:

```
Could not determine the install location of java2 package.
java2 package is required to run the license accept tool of this program.
Suggestion: install java2 and java2-jre packages provided by the operating
system install media.
```

You should install the java2 and java2-jre packages that can be obtained from the operating system installation media and run **new_install** again.

Troubleshooting for both RHEL3 and SLES9

- If you have any problem running **new_install** or **vac_configure**, you can use the following options to get a more verbose output:
 - v (verbose) provides some debug information.
 - vv (very verbose) provides a more detailed debug information than -v.
- In order to run **new_install** properly, you must have both gcc and gcc-c++ RPM packages installed so that their paths can be found by PATH environment variable. Otherwise when you run **new_install**, you may receive the following error message:

```
Could not determine location of gcc
```

or

```
Could not determine location of g++
```

If you do not want gcc and g++ to be found in the PATH environment variable, you can run **vac_configure** using your preferred gcc path by using the -gcc and -gcc64 manually. You can use the following command to determine if gcc and g++ can be found from PATH:

```
which gcc
```

For example, you may get the following output when you execute the above command:

```
/usr/bin/gcc
```

The output indicates that gcc can be found in one of the locations listed in the PATH environment variable. If the output is null, you must run **vac_configure** manually if you know the path to gcc/g++.

- If you have already installed the XL Fortran compiler, xlsmp.msg.rte, xlsmp.rte, xlsmp.lib and xlmass.lib packages have already been installed and RPM will prevent you from installing them for the second time. For example, if you have XL Fortran installed and try installing xlsmp and xlmass packages using the following command:

```
rpm -ivh xlsmp*.rpm xlmass*.rpm
```

You will get an error message similar to the following.

```
package xlsmp.lib-1.5.0-0 is already installed
package xlsmp.msg.rte-1.5.0-0 is already installed
package xlsmp.rte-1.5.0-0 is already installed
package xlmass.lib-4.1.0-0 is already installed
```

This is an expected message and you do not have to do anything to resolve it.

- If you try to install a package without installing its prerequisites, it will not be installed and you will get an error message. For example, if you try to install vac.cmp before xlsmp packages, vac.cmp will not be installed and the error message indicates the required packages as the prerequisite. Run the following command to install vac.cmp:

```
rpm -ivh vac.cmp-7.0.0-0.ppc64.rpm
```

If you do not have the xlsmp packages, you will get an error message similar to the following:

```
xlsmp.lib >= 1.5 is needed by vac.cmp-7.0.0-0
xlsmp.rte >= 1.5 is needed by vac.cmp-7.0.0-0
```

You should install the prerequisites first and try installing the package again. To avoid dependency errors during installation, install the compiler packages in the correct order mentioned in the default installation or non-default installation sections of this document.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Lab Director
IBM Canada Ltd. Laboratory
B3/KB7/8200/MKM
8200 Warden Avenue
Markham, Ontario L6G 1C7
Canada

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks and service marks

The following terms are trademarks of the International Business Machines Corporation in the United States, or other countries, or both:

- IBM
- iSeries
- pSeries

Other company, product, and service names may be trademarks or service marks of others.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.



Program Number: 5724-K77

GC09-7946-00

