IBM XL C for AIX, V12.1

Installation Guide

Version 12.1

GC14-7324-00

IBM XL C for AIX, V12.1

Installation Guide

Version 12.1

GC14-7324-00

Note
Before using this information and the product it supports, read the information in "Notices" on page 39.

First edition

This edition applies to IBM XL C for AIX, V12.1 (Program 5765-J01; 5725-C71) and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Contents

About this information

This guide provides detailed installation instructions for IBM® XL C for AIX®, V12.1. It guides you through multiple ways to perform tasks and directs you to reference information in cases requiring atypical installations. It also shows you how to test the installation, launch remotely-accessible HTML help, and enable and view different types of documentation. Please read it carefully before installing. Please also read the README file in the root directory of your installation media, which contains current information about the compiler.

Who should read this document

This document is intended for anyone responsible for installing IBM XL C for AIX, V12.1.

The basic installation methods detailed in this document address the needs of the majority of users. *Basic examples* are tailored to reflect, as much as possible, the steps required to perform a basic installation.

The advanced installation method addresses the needs of users who require a customized installation for various purposes, such as maintaining more than one version of XL C on a single system. These are users who are familiar with compiler installations and with the file structures of the versions of the compiler products installed on the system. *Advanced examples* are tailored to reflect, as much as possible, the steps required to perform an advanced installation.

How to use this document

This document provides procedures for three main installation scenarios:

"Basic" installation

This scenario allows you to install a single version of XL C to a default location. This scenario is applicable to the majority of users, and is the recommended method of installation.

For an overview of the steps that you need to follow to perform a basic installation, refer to Chapter 2, “Basic installation,” on page 9.

"Advanced" installation

This scenario allows you to maintain multiple versions of XL C on a single system, or to install the compiler to a non-default location. This scenario is only applicable to advanced users who have specialized needs, and is not recommended for the majority of users.

For an overview of the steps that you need to follow to perform an advanced installation, refer to Chapter 3, “Advanced installation,” on page 15.

"Update" installation

This scenario applies to users who have obtained a Program Temporary Fix (PTF) package for an existing XL C V12.1 installation.

For an overview of the steps required to install a PTF, refer to Chapter 4, “Update installation,” on page 21.

How this document is organized

This book is organized to reflect the pre-installation, installation, post-installation, and troubleshooting phases of an XL C installation.

Table 1. Phases of an XL C installation

Phase	Chapters	User segment
Pre-installation	Chapter 1, "Before installing XL C," on page 1	All users
Installation	Chapter 2, "Basic installation," on page 9	Users who: • Want to use the simplest, most direct installation process • Do not have any special requirements, such as the use of multiple versions of the compiler.
	Chapter 3, "Advanced installation," on page 15	Users who: • Want to install the compiler in a non-default location • Want to have multiple versions of the compiler on the same system
Post-installation	Chapter 4, "Update installation," on page 21	Users who want to update XL C V12.1 to the next fix level
	Chapter 5, "After installing XL C," on page 27	All users
Product removal	Chapter 6, "Uninstalling XL C," on page 35	Any user who needs to remove an XL C compiler from the system

Conventions

Typographical conventions

The following table explains the typographical conventions used in the IBM XL C for AIX, V12.1 information.

Table 2. Typographical conventions

Typeface	Indicates	Example
bold	Lowercase commands, executable names, compiler options, and directives.	The compiler provides basic invocation commands, **xlc**, along with several other compiler invocation commands to support various C language levels and compilation environments.
italics	Parameters or variables whose actual names or values are to be supplied by the user. Italics are also used to introduce new terms.	Make sure that you update the *size* parameter if you return more than the *size* requested.
underlining	The default setting of a parameter of a compiler option or directive.	nomaf | maf
`monospace`	Programming keywords and library functions, compiler builtins, examples of program code, command strings, or user-defined names.	To compile and optimize myprogram.c, enter: `xlc myprogram.c -O3`.

Qualifying elements (icons)

In descriptions of language elements where a feature is exclusive to the C1X standard, or where a feature is an IBM extension of the C standard, this information uses icons to delineate segments of text as follows:

Table 3. Qualifying elements

Qualifier/Icon	Meaning
IBM extension begins   IBM extension ends	The text describes a feature that is an IBM extension to the standard language specifications.
C1X, or C1X begins   C1X ends	The text describes a feature that is introduced into standard C as part of C1X.

Syntax diagrams

Throughout this information, diagrams illustrate XL C syntax. This section will help you to interpret and use those diagrams.

- Read the syntax diagrams from left to right, from top to bottom, following the path of the line.

  The ▸▸— symbol indicates the beginning of a command, directive, or statement.

  The —▸ symbol indicates that the command, directive, or statement syntax is continued on the next line.

  The ▸— symbol indicates that a command, directive, or statement is continued from the previous line.

  The —▸◂ symbol indicates the end of a command, directive, or statement.

  Fragments, which are diagrams of syntactical units other than complete commands, directives, or statements, start with the |— symbol and end with the —| symbol.
- Required items are shown on the horizontal line (the main path):

- Optional items are shown below the main path:

- If you can choose from two or more items, they are shown vertically, in a stack.

  If you *must* choose one of the items, one item of the stack is shown on the main path.

If choosing one of the items is optional, the entire stack is shown below the main path.

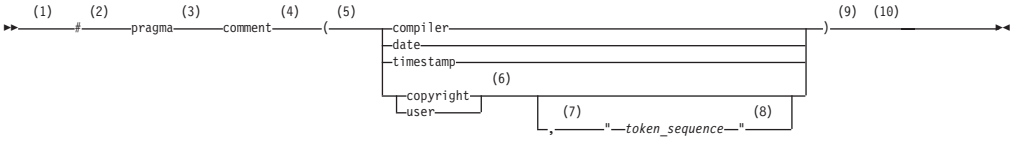
- An arrow returning to the left above the main line (a repeat arrow) indicates that you can make more than one choice from the stacked items or repeat an item. The separator character, if it is other than a blank, is also indicated:

- The item that is the default is shown above the main path.

- Keywords are shown in nonitalic letters and should be entered exactly as shown.
- Variables are shown in italicized lowercase letters. They represent user-supplied names or values.
- If punctuation marks, parentheses, arithmetic operators, or other such symbols are shown, you must enter them as part of the syntax.

Sample syntax diagram

The following syntax diagram example shows the syntax for the **#pragma comment** directive.



Notes:

1 This is the start of the syntax diagram.

2 The symbol # must appear first.

3 The keyword `pragma` must appear following the # symbol.

4 The name of the pragma `comment` must appear following the keyword `pragma`.

5 An opening parenthesis must be present.

6 The comment type must be entered only as one of the types indicated: `compiler`, `date`, `timestamp`, `copyright`, or `user`.

7 A comma must appear between the comment type `copyright` or `user`, and an optional character string.

8 A character string must follow the comma. The character string must be enclosed in double quotation marks.

9 A closing parenthesis is required.

10 This is the end of the syntax diagram.

The following examples of the **#pragma comment** directive are syntactically correct according to the diagram shown above:

```
#pragma comment(date)
#pragma comment(user)
#pragma comment(copyright,"This text will appear in the module")
```

Examples in this information

The examples in this information, except where otherwise noted, are coded in a simple style that does not try to conserve storage, check for errors, achieve fast performance, or demonstrate all possible methods to achieve a specific result.

The examples for installation information are labelled as either *Example* or *Basic example*. *Basic examples* are intended to document a procedure as it would be performed during a basic, or default, installation; these need little or no modification.

Related information

The following sections provide related information for XL C:

IBM XL C information

XL C provides product information in the following formats:

- README files

  README files contain late-breaking information, including changes and corrections to the product information. README files are located by default in the XL C directory and in the root directory of the installation CD.

- Installable man pages

  Man pages are provided for the compiler invocations and all command-line utilities provided with the product. Instructions for installing and accessing the man pages are provided in the *IBM XL C for AIX, V12.1 Installation Guide*.

- Information center

  The information center of searchable HTML files can be launched on a network and accessed remotely or locally. Instructions for installing and accessing the online information center are provided in the *IBM XL C for AIX, V12.1 Installation Guide*.

  The information center is viewable on the web at http://publib.boulder.ibm.com/infocenter/comphelp/v121v141/index.jsp.

- PDF documents

  PDF documents are located by default in the /usr/vac/doc/*LANG*/pdf/ directory, where *LANG* is one of en_US, zh_CN, or ja_JP. The PDF files are also available on the web at http://www.ibm.com/software/awdtools/xlc/aix/library/.

  The following files comprise the full set of XL C product information:

Table 4. XL C PDF files

Document title	PDF file name	Description
IBM XL C for AIX, V12.1 Installation Guide, GC14-7324-00	install.pdf	Contains information for installing XL C and configuring your environment for basic compilation and program execution.
Getting Started with IBM XL C for AIX, V12.1, SC14-7323-00	getstart.pdf	Contains an introduction to the XL C product, with information on setting up and configuring your environment, compiling and linking programs, and troubleshooting compilation errors.
IBM XL C for AIX, V12.1 Compiler Reference, SC14-7325-00	compiler.pdf	Contains information about the various compiler options, pragmas, macros, environment variables, and built-in functions, including those used for parallel processing.
IBM XL C for AIX, V12.1 Language Reference, SC14-7326-00	langref.pdf	Contains information about the C programming languages, as supported by IBM, including language extensions for portability and conformance to nonproprietary standards.
IBM XL C for AIX, V12.1 Optimization and Programming Guide, SC14-7327-00	proguide.pdf	Contains information on advanced programming topics, such as application porting, interlanguage calls with Fortran code, library development, application optimization and parallelization, and the XL C high-performance libraries.

To read a PDF file, use the Adobe Reader. If you do not have the Adobe Reader, you can download it (subject to license terms) from the Adobe website at http://www.adobe.com.

More information related to XL C including IBM Redbooks® publications, white papers, tutorials, and other articles, is available on the web at:

http://www.ibm.com/software/awdtools/xlc/aix/library/

For more information about boosting performance, productivity, and portability, see the C/C++ café at http://www.ibm.com/software/rational/cafe/community/ccpp.

Standards and specifications

XL C is designed to support the following standards and specifications. You can refer to these standards for precise definitions of some of the features found in this information.

- *Information Technology - Programming languages - C, ISO/IEC 9899:1990*, also known as *C89*.
- *Information Technology - Programming languages - C, ISO/IEC 9899:1999*, also known as *C99*.
- *Information Technology - Programming languages - Extensions for the programming language C to support new character data types, ISO/IEC DTR 19769*. This draft technical report has been accepted by the C standards committee, and is available at http://www.open-std.org/JTC1/SC22/WG14/www/docs/n1040.pdf.

- *AltiVec Technology Programming Interface Manual*, Motorola Inc. This specification for vector data types, to support vector processing technology, is available at http://www.freescale.com/files/32bit/doc/ref_manual/ALTIVECPIM.pdf.
- *Information Technology - Programming Languages - Extension for the programming language C to support decimal floating-point arithmetic, ISO/IEC WDTR 24732*. This draft technical report has been submitted to the C standards committee, and is available at http://www.open-std.org/JTC1/SC22/WG14/www/docs/n1176.pdf.
- *ANSI/IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Std 754-1985*.
- *OpenMP Application Program Interface Version 3.1*, available at http://www.openmp.org

Other IBM information

- *Parallel Environment for AIX: Operation and Use*
- The IBM Systems Information Center, at http://publib.boulder.ibm.com/infocenter/systems/index.jsp?topic=/com.ibm.aix.doc/doc/base/aixparent.htm is a resource for AIX information.

  You can find the following books for your specific AIX system:
  - *AIX Commands Reference, Volumes 1 - 6*
  - *Technical Reference: Base Operating System and Extensions, Volumes 1 & 2*
  - *AIX National Language Support Guide and Reference*
  - *AIX General Programming Concepts: Writing and Debugging Programs*
  - *AIX Assembler Language Reference*
- *ESSL for AIX V5.1/ESSL for Linux on POWER® V5.1 Guide and Reference* available at the Engineering and Scientific Subroutine Library (ESSL) and Parallel ESSL web page.

Other information

- *Using the GNU Compiler Collection* available at http://gcc.gnu.org/onlinedocs

Technical support

Additional technical support is available from the XL C Support page at http://www.ibm.com/software/awdtools/xlc/aix/support/. This page provides a portal with search capabilities to a large selection of Technotes and other support information.

If you cannot find what you need, you can send email to compinfo@ca.ibm.com.

For the latest information about XL C, visit the product information site at http://www.ibm.com/software/awdtools/xlc/aix/.

How to send your comments

Your feedback is important in helping to provide accurate and high-quality information. If you have any comments about this information or any other XL C information, send your comments by email to compinfo@ca.ibm.com.

Be sure to include the name of the information, the part number of the information, the version of XL C, and, if applicable, the specific location of the text you are commenting on (for example, a page number or table number).

Chapter 1. Before installing XL C

Before you install IBM XL C for AIX, V12.1:

- Familiarize yourself with the installation image, which contains the installable compiler filesets.
- Ensure that system prerequisites such as disk space and operating system are met.
- Determine the tasks you need to perform, depending on your installation requirements.
- Become either the root user or a user with administrator privileges.

Upgrading the evaluation copy of the XL C compiler

You can upgrade the evaluation version of the compiler to the licensed version by installing the licensed product package on top of the evaluation package. By following the Chapter 2, "Basic installation," on page 9 or Chapter 3, "Advanced installation," on page 15 procedure for the licensed product package, the compiler license can be installed. Shared filesets between the evaluation and licensed versions that have already been installed are reported as `Already installed`.

Note: If the evaluation version was installed using the advanced (non-default) installation method, and you want to install the licensed version using the same method, you must first remove the evaluation version using `rm -rf`, or install the licensed version to a different directory than the evaluation version.

The installation image and filesets

The installation image contains filesets for all the components of the compiler for all supported language locales.

Filesets

The image includes filesets for the compiler components.

This section lists the filesets that are included in the installation image and the default directories to which the fileset contents are installed during a basic installation. You might optionally install all available filesets, or only the ones you want. If you select specific filesets for installation, you might also choose to automatically install any prerequisite filesets by using the installp -g flag.

XL C compiler filesets

This topic lists the XL C compiler filesets V12.1.

Table 5. XL C compiler filesets and default installation locations

Fileset name	Fileset description	Default installation locations
vac.C	XL C compiler	/etc/ /usr/vac/ /usr/vac/bin/ /usr/vac/exe/ /usr/vac/listings/

Table 5. XL C compiler filesets and default installation locations (continued)

Fileset name	Fileset description	Default installation locations
vac.Bnd	XL C media defined bundles	/usr/sys/inst.data/sys_bundles/
vac.include	XL C compiler include files	/usr/vac/include/
vac.aix53	XL C libraries for AIX 5.3, AIX 6.1 and AIX 7.1	/usr/vac/lib/aix53/
vac.html.*LANG*	Supersede entry, not installed *LANG = [en_US, ja_JP, zh_CN]*	Not applicable
vac.html.common	Supersede entry, not installed	Not applicable
vac.lic	XL C license files	/usr/vac/lib/
vac.man.*LANG*	XL C compiler manual pages *LANG = [en_US, EN_US, ja_JP, JA_JP, Ja_JP, zh_CN, ZH_CN, Zh_CN]*	/usr/vac/man/*LANG*/man1/
vac.pdf.*LANG*.C	XL C compiler documentation (PDF) *LANG = [en_US, ja_JP, zh_CN]*	/usr/vac/doc/*LANG*/pdf/
vac.msg.*LANG*	XL C compiler messages *LANG = [en_US, EN_US, ja_JP, JA_JP, Ja_JP, zh_CN, ZH_CN]*	/usr/lib/nls/msg/*LANG*/
vac.ndi	XL C non-default installation script	/usr/vac/bin/

Runtime debug memory routine filesets

This topic lists the runtime debug memory routine filesets V5.4.

Table 6. Runtime debug memory routine filesets and default installation locations

Fileset name	Fileset description	Default installation locations
memdbg.adt	User heap/memory debug toolkit	/usr/vac/lib/ /usr/vac/lib/profiled/
memdbg.aix53.adt	User heap/memory debug toolkit for AIX 5.3, AIX 6.1, and AIX 7.1	/usr/vac/lib/aix53/ /usr/vac/lib/profiled/aix53/
memdbg.msg.*LANG*	User heap/memory debug messages *LANG = [en_US, EN_US, ja_JP, JA_JP, Ja_JP, zh_CN, ZH_CN]*	/usr/lib/nls/msg/*LANG*

Shared-memory parallelism (SMP) runtime environment filesets

This topic lists the SMP runtime filesets V3.1.

Table 7. SMP runtime filesets and default installation locations

Fileset name	Fileset description	Default installation locations
xlsmp.rte	SMP runtime library	/usr/include/ /usr/lpp/xlsmp/default_msg/
xlsmp.msg.*LANG*.rte	SMP runtime messages *LANG = [en_US, EN_US, ja_JP, JA_JP, Ja_JP, zh_CN, ZH_CN, Zh_CN]*	/usr/lib/nls/msg/*LANG*/

Table 7. SMP runtime filesets and default installation locations (continued)

Fileset name	Fileset description	Default installation locations
xlsmp.aix53.rte	SMP runtime libraries for AIX 5.3, AIX 6.1, and AIX 7.1	/usr/lpp/xlsmp/aix53/

Mathematical Acceleration Subsystem (MASS) filesets

This topic lists the MASS filesets V7.1.

Table 8. MASS filesets and default installation locations

Fileset name	Fileset description	Default installation locations
xlmass.adt.include	IBM MASS application development include files	/usr/xlmass/include/
xlmass.lib	IBM MASS libraries	/usr/xlmass/lib/
xlmass.aix53.lib	IBM MASS libraries for AIX 5.3, AIX 6.1, and AIX 7.1	/usr/xlmass/lib/aix53/

IBM Debugger filesets

This topic lists the IBM Debugger filesets V12.1.

Table 9. IBM Debugger filesets

Fileset name	Fileset description
ibmdebugger	IBM Debugger for AIX
ibmdebugger.engine	IBM Debugger for AIX Compiled Languages Engine
ibmdebugger.engine.msg.*LANG*	IBM Debugger messages *LANG = [de_DE, en_US, en_ES, fr_FR, it_IT, ja_JP, ko_KR, pt_BR, zh_CN, zh_TW]*
ibmdebugger.jre	Supersede entry, not installed
ibmdebugger.ui	Supersede entry, not installed

The IBM Debugger for AIX, V12.1 is included with the XL C installation image. This source-level debugger uses a client/server model:

- The server is a debugger engine that is installed on AIX.
- The client user interface is an Eclipse-based debugger that is installed on Windows workstations.

Notes:

- IBM Debugger for AIX, V12.1 is a 64-bit application. It cannot be used on a 32-bit machine.
- IBM Debugger for AIX, V12.1 supersedes IBM Distributed Debugger. If the IBM Distributed Debugger V9.2 is installed on your system, it will be removed upon installation of the IBM Debugger for AIX, V12.1. The post-installation script then modifies the idebug fileset levels in the Object Data Manager (ODM) database to 99.99.9999.9999 to prevent future installations of idebug.* filesets.
- To install IBM Debugger and the Windows remote debug interface, see "README.debug" in the root directory of the installation image.

Utilization reporting tool fileset

This topic lists the utilization reporting tool filesets V1.1.

Table 10. Utilization reporting tool fileset and default installation locations

Fileset name	Fileset description	Default installation locations
urt11	Utilization reporting tool files *LANG = [en_US, EN_US, ja_JP, JA_JP, Ja_JP, zh_CN, ZH_CN]*	/opt/ibmurt/1.1/ /opt/ibmurt/1.1/bin/ /opt/ibmurt/1.1/man/*LANG*/man1/ /opt/ibmurt/1.1/msg/*LANG*/ /opt/ibmurt/1.1/config/

Note: The utilization reporting tool can be installed on the same machine as the compiler or on a different machine. You must use **installp** utilities to install `urt11` fileset. It typically needs to be installed only once, no matter how many compiler installations you have. This tool is useful for those who are responsible for monitoring the overall compiler usage within an organization. Individual developers and build teams should not need to use this tool.
To install the utilization reporting tool, see the instructions provided in `README.urt` in the root directory of the installation image.

Preparing the electronic distribution package

If you have downloaded the XL C electronic distribution package, xlc.12.1.0.aix.GM.tar.Z, follow the steps described in this topic before the installation.

Note: You must be logged in as root user or have superuser authority.
To uncompress and unpack the file, use the command:

```
zcat xlc.12.1.0.aix.GM.tar.Z | tar -xvf -
```

The following example shows the steps to prepare the electronic distribution package, xlc.12.1.0.aix.GM.tar.Z, located in the new directory, `/compiler/install/` for installation.

```
cd /compiler/install
zcat xlc.12.1.0.aix.GM.tar.Z | tar -xvf -
cd usr/sys/inst.images
```

See "System prerequisites" on page 5 to ensure that system prerequisites are met and use the installation methods described in either Chapter 2, "Basic installation," on page 9 or Chapter 3, "Advanced installation," on page 15 to install the compiler.

National language support

When fileset names differ only by the locale code, you can choose to install only the filesets that are relevant to your desired language and location. The LANG environment variable determines which message catalogs are used. The en_US (English) message catalogs are installed by default. If LANG is not defined or is assigned an unsupported locale, en_US message catalogs are used. This topic introduces a list of available locales.

The XL C messages support the following language locales:

- en_US (ISO8859-1)
- EN_US (UTF-8)
- ja_JP (IBM-eucJP)

- JA_JP (UTF-8)
- Ja_JP (IBM-943)
- zh_CN (IBM-eucCN)
- ZH_CN (UTF-8)
- Zh_CN (GB18030)

English (en_US) is the default national language. Following installation, you can set the NLSPATH so that messages are displayed in a different language. See "Enabling the XL C error messages" on page 29.

System prerequisites

All system prerequisites must be ready on your system before XL C and its documentation can be successfully installed and run.

- **Supported operating systems:**
  - AIX V5.3 TL 5300-07 or later
  - AIX V6.1
  - AIX V7.1
  - IBM i V6.1 PASE V6.1 with PTF SI30636 or later
  - IBM i V7.1 PASE V7.1

  Note: Additional filesets must be installed on these systems. For details, see "Required filesets for installing XL C."
- **Required hard disk space:** 430 MB
- **Required software for documentation:**
  - HTML browser
  - PDF viewer

Verifying the amount of hard disk space available

You can use the following command to determine the amount of space available in the default installation location (/usr/vac/):

```
df -m /usr
```

If you plan to install the compiler to a non-default location, you can use the following command:

```
df -m installation_path
```

In this command, *installation_path* represents the non-default location.

Required filesets for installing XL C

The following filesets must be installed on your system in order to install IBM XL C for AIX, V12.1:

Fileset name	Fileset description
bos.adt.include	Base application development include files
bos.adt.lib	Base application development libraries
bos.adt.libm	Base application development math libraries
bos.rte	Base operating system run time

Fileset name	Fileset description
bos.rte.libc	Base application runtime library

At least one of the following locale-specific filesets must be installed on your system:

Fileset name	Fileset description
bos.loc.*LANG*	Base locale support
bos.loc.pc.*LANG*	Base system locale PC code set
bos.loc.utf.*LANG*	Base system locale UTF code set
bos.loc.iso.*LANG*	Base system locale ISO code set

You can view information about the installed filesets by using the **lslpp** command. For example, to determine if the required filesets have been installed, use the following command:

```
lslpp -L bos.adt.include bos.adt.lib bos.adt.libm bos.loc.* bos.rte bos.rte.libc
```

Note: If one of the required filesets is missing, an error message might be generated, for example,

```
MISSING REQUISITES: The following filesets are required by one or more of the selected
filesets listed above. They are not currently installed and could not be found on the
installation media.

bos.adt.libm 5.3.7.0 # Base Level Fileset
```

This message indicates that the `bos.adt.libm` fileset is required but not installed.

To install a missing fileset, use the following command:

```
# installp -acXgd installation_path fileset
```

In this command, *installation_path* represents the location of the fileset to be installed. For further information, see the installation video on IBM Education Assistant at http://publib.boulder.ibm.com/infocenter/ieduasst/rtnv1r0/topic/com.ibm.iea.compileraix/compileraix/12.1/Installation/Prereq_installation/libm_installation_viewlet_swf.html?dmuid=20111028142719997096.

Previewing the installation and license agreements

Before installing IBM XL C for AIX, V12.1, you can verify the default installation process, check file sizes and disk space, and preview the license agreements using either SMIT or installp.

The terms and conditions for using XL C are specified in the following PDF files, which are located in the root directory of the distribution package:

- LicenseAgreement.pdf
- license.pdf

These license files can be found in the /usr/vac/ directory after basic installation.

Note: You can use the utilization tracking and reporting feature to ensure the compiler usage is compliant with the number of Concurrent User licenses you have purchased. For more information, see Tracking and reporting compiler usage in the *XL C Compiler Reference*.

Using SMIT

You can use the System Management Interface Tool (SMIT) to preview the basic XL C installation.

Follow the steps to install the compiler to the default location given in "Using SMIT" on page 11. When prompted with the SMIT dialog to customize the installation in step 3, set the `Preview only?` field to **yes**. With this setting enabled, SMIT displays progress messages at each stage of the installation but does not actually install the compiler on your system.

Using installp

You can also preview the command line installation process and write the output to a log file, using the **installp** command. Follow the steps to install the compiler to the default location given in "Using installp" on page 12 and specify the **-p** flag to preview the process without installing the compiler.

The following example previews the installation of all the available compiler filesets in the `/cdrom/usr/sys/inst.images` source directory to the default location, /usr/vac/ and writes an installation log file to `/tmp/install.log`.

Example:

```
installp -paXgd /cdrom/usr/sys/inst.images -e /tmp/install.log all
```

Chapter 2. Basic installation

You can use the basic installation procedure to install XL C if the following conditions are true:

- You are installing IBM XL C for AIX, V12.1 to the default directory, /usr/vac/.
- You are maintaining a single version of XL C on your system, and you agree to remove any previously installed XL C components.

If both these conditions apply, the basic installation is the easiest and fastest method, because it can automatically uninstall previous versions of XL C, install the latest version, and configure the compiler, all through the use of a single installation tool.

If either of these conditions does not apply, do not use basic installation. Instead, see the procedures in Chapter 3, "Advanced installation," on page 15.

The following basic installation methods install XL C to the default directory. If you have root access or a profile with superuser privileges, you can use the Bundle installation method, the System Management Interface Tool (SMIT), or the **installp** command to install the compiler.

You can use the basic installation methods to perform any of the following tasks:

- Install IBM XL C for AIX, V12.1 on a clean system.
- Install IBM XL C for AIX, V12.1 on a system where IBM XL Fortran for AIX, V14.1 is already installed. In this case, both compilers will use the SMP and MASS libraries provided with XL C for AIX. (For information about installing XL Fortran, see the *IBM XL Fortran for AIX, V14.1 Installation Guide*.)
- Install an update on a system where IBM XL C for AIX, V12.1 has already been installed. In this case, use the procedure in Chapter 4, "Update installation," on page 21.

Installation demonstration videos are also available on IBM Education Assistant at http://publib.boulder.ibm.com/infocenter/ieduasst/rtnv1r0/index.jsp. For basic installation instructions and how to upgrade the evaluation copy, see http://publib.boulder.ibm.com/infocenter/ieduasst/rtnv1r0/topic/com.ibm.iea.compileraix/compileraix/12.1/Installation.html?dmuid=20111028143138027594.

Tasks for basic installation

Before you perform a basic installation, you must make sure that you have a user account with administrator privileges and that your system meets all system requirements. After you have finished installing the compiler, you must confirm that the installation has been successful and enable the man pages of the compiler.

The following table outlines the procedure for basic installation.

Table 11. Steps for basic installation: new or upgrade installation

Steps	References
Become either the root user or a user with administrator privileges.	Documentation supplied with the operating system.

Table 11. Steps for basic installation: new or upgrade installation (continued)

Steps	References
Ensure that all system prerequisites are satisfied.	"System prerequisites" on page 5
Use the bundle installation method, SMIT, or the **installp** command to install and configure the compiler, using the default paths.	"Using the bundle installation method," "Using SMIT" on page 11, or "Using installp" on page 12
Confirm that the compiler packages have been successfully installed, and test the installation.	"Checking compiler and fileset information" on page 27, "Testing the installation" on page 33
Enable the compiler man pages.	"Enabling the manual pages" on page 30
If your system locale or encoding, or both are not en_US, enable the compile-time error messages. Otherwise, skip this step.	"Enabling the XL C error messages" on page 29

Using the bundle installation method

You can install XL C using the bundle installation method. The bundle fileset contains scripts to help install either the evaluation version or the licensed version of the compiler.

About this task

To install the compiler using the bundle installation method, take the following steps:

Procedure

1. As the root user, install the compiler bundle fileset `vac.Bnd`, using the **installp** command. You can skip this step if you are upgrading from the evaluation version and the fileset is already installed on the system.

   ```
   installp -aXd . vac.Bnd
   ```

2. Enter the following command from the command line to start Smit easy mode:

   ```
   # smitty easy_install
   ```

3. Enter the name of the installation device. It is the path in which the filesets can be found, for example, `/usr/sys/inst.images`. If the filesets are in your current directory, you can use "." as the input device.

   ```
   * INPUT device / directory for software    [.]
   ```

4. From the selection screen, select the package that you need.
   - If you are installing or upgrading to the licensed version on AIX V5.3, choose `vac-licensed-aix53`.
   - If you are installing or upgrading to the licensed version on AIX V6.1 or V7.1, choose `vac-licensed-aix61-aix71`.
5. Change the following values provided on the bundle screen as appropriate to your situation, and press **Enter** to continue.
   - You can change the **AUTOMATICALLY install requisite software?** option to yes to install the compiler automatically.
   - You can change the **EXTEND file systems if space needed?** option to yes to extend file systems.
   - You must change the **ACCEPT new license agreements?** option to yes to accept new license agreements.

6. Scroll to the end of the installation output to check the installation summary. The output indicates whether the installation of your bundle has been successful. Exit using Esc+0.

   Note: Using the bundle installation method prevents the installation tools from installing filesets that are not supported by your operating system. Therefore, no failures and error messages caused by unsupported filesets are generated as mentioned in "Using SMIT" and "Using installp" on page 12.

Using SMIT

The SMIT guides you step-by-step through the installation process.

About this task

To install the compiler using the SMIT interface, follow these steps:

Procedure

1. Enter the command `smit install_latest`. This command runs the SMIT, which presents a menu-driven user interface for the installation process. You must specify the `INPUT device/directory of the software`.
2. Press **F4** to select from a list of suggested devices and directories, or type in the name of the CD-ROM device or the path to the location of the packages. If you want to install over a network you must have a network server installed, and then specify the directory on the client that corresponds to the installation source on the network server.
3. Press **Enter**. At this stage you are prompted with a list of questions that can help you customize the compiler installation on your system. Verify the default values, or modify them where necessary to accommodate your preferences.

   Important: You must select **yes** next to `Accept new LICENSE agreements?` to continue with the installation.

   Note:

   - You can select the `SOFTWARE to install` field and press **F4** to view the installable filesets on the device or in the directory, then select the filesets to install. Install the runtime libraries first, then install the other filesets in any order. It is recommended that you keep the default setting, `_all_latest`, to install all the filesets.
   - To preview the installation process without actually installing the compiler select **yes** next to the `Preview only?` field.
4. Press **Enter** and then **Enter** again.

   Notes:

   - If you use the command `smit install_latest`, an error message might be generated. For example, you might encounter an error message indicating that either `xlC.aix50.rte` or `xlC.aix61.rte` has failed to be installed. In this case, you can ignore the message. The reason for the error message is that if you install the compiler onto a system running AIX 5.3, only `xlC.aix50.rte` is required, and `xlC.aix61.rte` automatically fails to be installed. Likewise, if you install the compiler onto a system running AIX 6.1, only `xlC.aix61.rte` is required, and `xlC.aix50.rte` fails to be installed. The following message is an example of the error message you might encounter.

```
Requisite Failures
------------------
SELECTED FILESETS: The following is a list of filesets that you asked to
install. They cannot be installed until all of their requisite filesets
are also installed. See subsequent lists for details of requisites.

xlC.aix61.rte 11.1.0.0 # XL C/C++ Runtime for AIX 6.1

MISSING REQUISITES: The following filesets are required by one or more
of the selected filesets listed above. They are not currently installed
and could not be found on the installation media.

bos.rte 6.0.0.0 # Base Level Fileset
bos.rte.libc 6.0.0.0 # Base Level Fileset
```

- Similarly, any language specific filesets that are not supported by your system can produce an error message and fail to install. For example, if local support is not present for Ja_JP, you might fail when attempting to install a Ja_JP specific language fileset for the compiler. These error messages are normal and must be ignored.
- The runtime libraries and compiler filesets are always required.
- You do not need all of the compile-time and runtime message catalogs, only those for the national languages that you need for compile-time and runtime messages.
- Messages are displayed as each part is successfully installed.
- The installation log is stored in the `/home/smit.log` file.

Using installp

You can install XL C directly from the command line using the AIX command `installp`.

About this task

To install the compiler and write the installation output to a log file using the **installp** command, follow these steps:

Run the command:

```
installp -aXYgd install_images_location -e logfile fileset_names
```

-aYg
Specifies that all the latest installable filesets available in the *install_images_location* directory are applied and that the required software license agreement is accepted.

-p
Performs a preview of the installation process by running the preinstallation checks. See previewing the installation.

-X
Attempts to expand the file system at the default location if there is insufficient space to complete the install.

-d *install_images_location*
Specifies the directory where the filesets are located. This path may also be a mounted CD-ROM drive.

-e *logfile*
Specifies the name and location of the installation log file.

fileset_names
Specifies a list of names of the filesets in *install_images_location* that you want to install.

Note: The keyword *all* can be used to indicate that all filesets should be attempted during installation.

The following example installs all available compiler filesets in the `/cdrom/usr/sys/inst.images` source directory to the default locations, as specified in "Filesets" on page 1, and writes an installation log file to `/tmp/install.log`.

Example:

```
installp -aXYgd /cdrom/usr/sys/inst.images -e /tmp/install.log all
```

Notes:

- If you use the keyword `all` as in this example, an error message might be generated. For example, you might encounter an error message indicating that either `xlC.aix50.rte` or `xlC.aix61.rte` has failed to be installed. In this case, you can ignore the message. The reason for the error message is that if you install the compiler onto a system running AIX 5.3, only `xlC.aix50.rte` is required, and `xlC.aix61.rte` automatically fails to be installed. Likewise, if you install the compiler onto a system running AIX 6.1, only `xlC.aix61.rte` is required, and `xlC.aix50.rte` fails to be installed. The following message is an example of the error message you might encounter.

  ```
  Requisite Failures
  ------------------
  SELECTED FILESETS: The following is a list of filesets that you asked to
  install. They cannot be installed until all of their requisite filesets
  are also installed. See subsequent lists for details of requisites.

  xlC.aix61.rte 11.1.0.0 # XL C/C++ Runtime for AIX 6.1

  MISSING REQUISITES: The following filesets are required by one or more
  of the selected filesets listed above. They are not currently installed
  and could not be found on the installation media.

  bos.rte 6.0.0.0 # Base Level Fileset
  bos.rte.libc 6.0.0.0 # Base Level Fileset
  ```

- Similarly, any language specific filesets that are not supported by your system will produce an error message and fail to install. For example, if locale support is not present for Ja_JP, you will get a failure when attempting to install a Ja_JP specific language fileset for the compiler. These error messages are normal and should be ignored.

Chapter 3. Advanced installation

You must use the advanced installation procedure if any of the following are true:

- You are maintaining multiple versions of the same product on a single system.
- You are installing the product to a non-default location.
- You are installing on an IBM i V6.1 PASE system.
- You are installing on an IBM i V7.1 PASE system.

To perform an advanced installation, you need to use the `vacndi` Perl script provided with the installation image, to install XL C to a non-default directory and run multiple versions of XL C on a single system. The script is packaged in the vac.ndi fileset. To avoid unexpected behavior during installation, do not modify this script.

Limitations of non-default installation

Although non-default installation gives you more freedom to choose how many versions of XL C to keep on your system and where to install them, in many ways a non-default installation is also more restrictive, demanding to implement, and difficult to maintain than a basic installation.

- The recommended method for installing XL C is installing to the default location. You must only use the `vacndi` script to install XL C if you are an expert user familiar with the compiler and the standard installation process.
- The `vacndi` script does not check whether prerequisite packages are installed.
- You cannot use this script to install the product filesets individually.
- You cannot use AIX tools (such as **`lslpp`**) to uninstall or to query installed filesets for version or release level information.
- Service refreshes of the compiler assume a default installation path. For a non-default compiler installation, the `vacndi` script must be used to apply service packages.
- To run `vacndi`, you must have Perl version 5.5.3 or higher, runtime environment `perl.rte`, installed on your system. This fileset is shipped with the AIX base operating system.
- The `vacndi` script requires that the original filesets not be renamed. This means that you cannot install the compiler using filesets that have been copied by the SMIT sub-option "Copy Software to Hard Disk for Future Installation" because the version number of the compiler is appended to the filesets when they are copied.
- The `vacndi` tool requires you to install the PTF vac.ndi fileset before the tool can be used to install the PTF to a non-default location.

Tasks for advanced installation

Before you perform an advanced installation, you must make sure that you have a user account with administrator privileges and that your system meets all system requirements. You might also want to uninstall any previous versions of XL C on your system. After you have finished installing the compiler you must confirm that the installation has been successful and enable the man pages of the compiler.

The recommended procedure for advanced installation is outlined in the following table:

Table 12. Steps for advanced installation: new or upgrade installation

Task	For more information, see . . .
Become either the root user or a user with administrator privileges.	Documentation supplied with the operating system.
Ensure that all system prerequisites are satisfied.	"System prerequisites" on page 5
If you do not need to maintain multiple versions of the product on your system, remove any existing versions of XL C for AIX.	Chapter 6, "Uninstalling XL C," on page 35
Use the non-default installation script to install the compiler. If you are installing the compiler on an IBM PASE system, use the non-default installation method.	"Non-default installation procedure" "Installing on an IBM i PASE system" on page 18
Confirm that the compiler packages have been successfully installed, and test the installation.	"Checking compiler and fileset information" on page 27, "Testing the installation" on page 33
Enable the compiler man pages.	"Enabling the manual pages" on page 30
If your system locale or encoding, or both are not en_US, enable the compile-time error messages. Otherwise, you can skip this step.	"Enabling the XL C error messages" on page 29

Non-default installation procedure

To install XL C to a non-default location, first use installp to install the vac.ndi fileset, then use the `vacndi` Perl script included in that fileset to install the compiler.

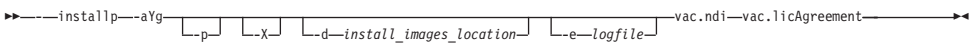
About this task

Follow these steps to install XL C to an alternate location:

Procedure

1. First use the **installp** command to install the vac.ndi fileset. Run the command:

   ```
   installp -aYgd install_images_location -e logfile vac.ndi vac.licAgreement
   ```

   -aYg
   Specifies that all the latest installable filesets available in the *install_images_location* directory are applied and that the required software license agreement is accepted.

   -d *install_images_location*
   Specifies the directory where the filesets are located. This path might also be a mounted CD-ROM drive.

   -e *logfile*
   Specifies the name and location of the installation log file. By default, the installation log file vacndi.log will be stored in your working directory.

-p
Performs a preview of the installation process by running the preinstallation checks. See "Previewing the installation and license agreements" on page 6.

-X
Attempts to expand the file system at the default location if there is insufficient space to complete the install.

Example:

```
installp -aYgd /cdrom/usr/sys/inst.images -e /tmp/install.log vac.ndi \
    vac.licAgreement
```

Note: This command automatically installs and accepts the license agreement fileset, vac.licAgreement, required to install vac.ndi.

2. Install any available updates to the vac.ndi fileset. You can download the latest compiler updates from the support website: http://www.ibm.com/software/awdtools/xlc/aix/support/

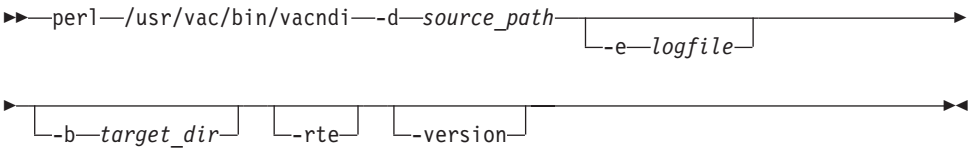
   To check the version number of the vac.ndi fileset currently installed on your system, use the following command:

   ```
   lslpp -l vac.ndi
   ```

   To install a new version of the vac.ndi fileset, use the following command:

   ```
   installp -aYgd ptf_images_location -e logfile vac.ndi
   ```

3. Install XL C by entering the following command:




```
perl /usr/vac/bin/vacndi -d source_path [-e logfile] [-b target_dir] [-rte]
[-version]
```

where:

-d *source_path*
Specifies the directory where the filesets are located. This path may also be a mounted CD-ROM drive.

-e *logfile*
Specifies the name and location of the installation log file. By default, the installation log file vacndi.log is stored in your working directory.

-b *target_dir*
Specifies the location where the filesets should be copied and expanded. By default, the files are copied to the vacndi directory in your working directory. If the directory exists already, you will receive an error message and the installation will stop.

-rte
Specifies that only the runtime component should be installed.

-version
Displays the version of the non-default installation tool.

The following example uses the `vacndi` Perl script to install all available compiler filesets in the `/cdrom/usr/sys/inst.images` source directory to the non-default location, `/compiler/xlc/` and writes the installation log file to `/tmp/xlc.ndi.inst.log`.

Example:

```
perl /usr/vac/bin/vacndi -d /cdrom/usr/sys/inst.images -b /compiler/xlc/
-e /tmp/xlc.ndi.inst.log
```

Results

IBM XL C for AIX is installed into the chosen location.

Installing on an IBM i PASE system

About this task

Users installing XL C on an IBM i PASE system must use the non-default installation method because the AIX SMIT and installp utilities are not supported.

If, during installation, the `vacndi` script determines that you are installing the compiler onto a PASE system, it modifies the name of the XL C configuration file and one of its variables, oslevel, to match the version of PASE you are running. For example, if the compiler has been installed on a system running IBM i V6.1 PASE V6.1, vac.cfg.53 is renamed vac.cfg.61 and the value of oslevel is changed from 5.3 to 6.1. If the compiler has been installed on a system running IBM i V7.1 PASE V7.1, vac.cfg.61 is renamed vac.cfg.71 and the value of oslevel is changed from 6.1 to 7.1.

To install the XL C compiler on i PASE, follow these steps:

1. Start the installation by inserting the compiler product installation CD into the CD drive, or using the electronic distribution package.

   Note: If you use the electronic image, you must extract the distribution package into a case sensitive file system, such as `/QOpenSys/`. The non-default installation option **-i** can only be used if you install the compiler using the CD.
2. Sign on to the system with a user profile that has *ALLOBJ authority. The compiler product files are owned by this user profile.
3. Start an interactive i PASE terminal session by entering this CL command.

   ```
   call qp2term
   ```
4. Restore the appropriate compiler installation script.
   - If you use the electronic distribution package, run the following commands:

     ```
     cd /QOpenSys/home/user
     zcat xlc.12.1.0.aix.GM.tar | tar -xvf -
     restore -qf usr/sys/inst.images/vac.ndi ./usr/vac/bin/vacndi
     ```
   - If you use the installation CD, run the following commands:

     ```
     cd /QOpenSys/home/user
     restore -qf /QOPT/CDROM/USR/SYS/INST.IMA/VAC.NDI ./usr/vac/bin/vacndi
     ```
5. Run the installation script to install the compiler. The destination directory for the compiler is specified by the **-b** option in the command. The preferred directory names for the compilers are used in the following commands. If you choose a different directory, the directory is in the `/QOpenSys` tree to allow for case-sensitive file names.
   - If you use the electronic distribution package, run the following commands:

```
/QIBM/ProdData/DeveloperTools/pase/bin/perl ./usr/vac/bin/vacndi -d \
./usr/sys/inst.images -b /QOpenSys/xlc
```

- If you use the installation CD, run the following commands:

```
/QIBM/ProdData/DeveloperTools/pase/bin/perl ./usr/vac/bin/vacndi -i -d \
/QOPT/CDROM/USR/SYS/INST.IMA -b /QOpenSys/xlc
```

After the XL C compiler is installed for use in i PASE,

- The XL C compiler commands, for example, **xlc, cc** can be found in directory `/QOpenSys/xlc/usr/vac/bin/`.
- The XL C compiler documentation (in PDF) can be found in directory `/QOpenSys/xlc/usr/vac/doc/en_US/pdf`.

Note: You can add the directory that contains the compiler commands to the *$PATH* environment variable. The *$PATH* environment variable specifies the directories to be searched to find a command. For example, if you have installed XL C/C++ compiler, you can change the *$PATH* environment variable as follows to avoid specifying the command path when you use the compiler commands:

```
export $PATH=$PATH:/QOpenSys/xlc/usr/vac/bin
```

For information about installing PTFs for compiler installation updates on the PASE system, see "Installing PTF updates on the PASE system" on page 22.

For more information about the IBM i V7.1 PASE system, see IBM PASE for i. For more information about the IBM i V6.1 PASE system, see i5/OS® PASE, or see the PDF version of *System i®: Programming IBM PASE for i*.

Running XL C from a non-default location

The compiler invocation commands are installed to `/target_dir/usr/vac/bin/`, where `target_dir` is the location of the compiler as set by the **-b** flag during the non-default installation.

About this task

For example, if you installed XL C to the `/compiler/xlc` directory, you can run the compiler using:

```
/compiler/xlc/usr/vac/bin/xlc -o hello_c hello.c
```

Alternatively, you can create a shortcut to the compiler invocations, as described in "Creating shortcuts to the XL C invocation commands" on page 31.

Chapter 4. Update installation

A program temporary fix (PTF) is an update to XL C that provides a fix or multiple fixes to the product.

You can download the latest compiler updates from the support website: http://www.ibm.com/software/awdtools/xlc/aix/support/ . You can also find the instructions on Fix Central.

Tasks for update installation

If you are installing an update to XL C, you must first determine whether the compiler has been installed to a default or non-default directory, and then follow the recommended steps for updating that installation type.

If the compiler has been installed to the default directory, /usr/vac/, you must follow the steps outlined in the following table, Table 13.

Table 13. Steps for basic installation: update installation

Task	**For more information, see . . .**
Become either the root user or a user with administrator privileges.	Documentation supplied with the operating system.
Ensure that all the system prerequisites are met.	"System prerequisites" on page 5
Download, uncompress, and unpack the PTF package.	"Preparing a PTF package for installation" on page 22
Generate a .toc file.	"Preparing a PTF package for installation" on page 22
Use SMIT or the **installp** command to install PTFs, using the default paths.	"Using SMIT" on page 23 or "Using installp" on page 24
Confirm that the compiler packages were successfully installed, and test the installation.	"Checking compiler and fileset information" on page 27, "Testing the installation" on page 33

If the compiler has been installed to any directory other than /usr/vac/, you must follow the steps outlined in the following table, Table 14.

Table 14. Steps for advanced installation: update installation

Task	**For more information, see . . .**
Become either the root user or a user with administrator privileges.	Documentation supplied with the operating system.
Ensure that all system prerequisites are met.	"System prerequisites" on page 5
Download, uncompress, and unpack the PTF package.	"Preparing a PTF package for installation" on page 22
Generate a .toc file.	"Preparing a PTF package for installation" on page 22
Install any PTF updates to the vac.ndi fileset using SMIT or installp.	"Using SMIT" on page 23 or "Using installp" on page 24
Use the non-default installation script to install further updates.	"Updating an advanced installation using vacndi" on page 24

Table 14. Steps for advanced installation: update installation (continued)

Task	For more information, see . . .
Confirm that the compiler packages were successfully installed, and test the installation.	"Checking compiler and fileset information" on page 27, "Testing the installation" on page 33

Preparing a PTF package for installation

Updates to the compiler are supplied as PTF packages. Before these PTF packages can be used to update the compiler, they must be downloaded and uncompressed, and a .toc file must be generated.

Use the following instructions to prepare the update package for installation:

Note: You must be logged in as root or have superuser authority.

1. Download the PTF package that you want(for example `vac.121.aix.mthyear.ptf.tar.Z`) into an empty directory.
2. Restore the compressed file and extract the PTF filesets from the downloaded package. To uncompress and unpack the tar file, use the following command:
   ```
   zcat package_name.tar.Z | tar -xvf -
   ```
3. Use the command **inutoc directory_name** to generate a .toc file in the same directory as the unpacked filesets.

The following example shows the steps to prepare the package, `vac.121.aix.mthyear.ptf.tar.Z`, located in the new directory, `/compiler/update/` for installation.

```
cd /compiler/update
zcat vac.121.aix.mthyear.ptf.tar.Z | tar -xvf -
inutoc .
```

Installing PTF updates on the PASE system

This section describes the steps to install PTFs for the XL C compiler on the PASE system.

Before installing the PTFs, you must have already installed the compilers on the PASE system using the steps in "Installing on an IBM i PASE system" on page 18. To install PTFs for the XL C compiler on i PASE, follow these steps:

1. Obtain the PTF package files to be installed.
2. Uncompress and then untar the PTF package files. If you have downloaded the compressed TAR images to the `/QOpenSys/vacptf/` directory, you can use the following commands from a QP2TERM command line:
   ```
   cd /QOpenSys/ptf
   uncompress <filename.tar.Z>
   tar -xvf <filename.tar>
   ```
3. Create a file containing a list of the PTF packages to be installed. To do so, use the following commands on a QP2TERM command line:
   ```
   cd /QOpenSys/ptf
   ls *.bff > ptflist.txt
   ```
4. Check and update, if necessary, the non-default installation (NDI) tool itself. The updated version of the NDI tool needs to be restored first in order to use the updated version to install the rest of the PTF package. To do so, use the following commands:

```
cd /
ls /QOpenSys/ptf/vac.ndi.*
restore -qf vac.ndi.V.R.M.F.bff ./usr/vac/bin/vacndi
```

Note:

- If the **ls** command returns an error message indicating the file does not exist, it means the NDI tool is not updated by the PTF package. Skip this command and continue with the next step in the instructions.
- If the **ls** command does return a file name, make a note of this file name and use that file name in the **restore** command. For example, if the file name returned is `vac.ndi.12.00.0000.0006.bff`, then replacing the `vac.ndi.V.R.M.F.bff` file name shown in the **restore** command with the actual name returned results in the following **restore** command:

  ```
  restore -qf vac.ndi.12.00.0000.0006.bff ./usr/vac/bin/vacndi
  ```

5. Run the installation script to install the PTFs. Enter the following command from the QP2TERM command line:

   ```
   /QIBM/ProdData/DeveloperTools/pase/bin/perl /usr/vac/bin/vacndi \
    /QOpenSys/ptf -b /QOpenSys/xlc -u /QOpenSys/ptf/ptflist.txt
   ```

The installation script creates a compressed TAR backup of the compiler files that exist before the PTF update. If you use the directories as shown in these instructions, this file is named `/QOpensys/xlc.backup.tar.Z`. If a problem is encountered with the installation of the PTF update or with the PTF update itself, you can restore from this backup to uninstall the PTF update.

Updating a basic installation

If you used a basic installation process to install XL C to a default location, it is recommended that you use SMIT or the **installp** command to install PTFs.

Using SMIT

About this task

To install a PTF using the SMIT interface, follow these steps:

Procedure

1. Run the command: `smit install_all`. This command invokes the SMIT, which presents a menu-driven user interface for the installation process.
2. When asked to, specify the `INPUT device/directory of the software`, enter the path of the directory where the PTF filesets and .toc are located.
3. Press **Enter**.

   At this stage, you are prompted with a list of questions that will help you customize the compiler installation on your system. Verify the default values, or modify them where necessary, to accommodate your preferences.
4. Press **Enter** and then **Enter** again.

   Messages are displayed as each part is successfully installed, and the installation log is stored to `/home/smit.log`.

What to do next

Note: PTF packages often ship base level filesets as prerequisites. If the command `smit update_all` is used, then those packages will not install, which may cause

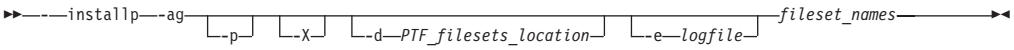
requisite failures. For this reason we recommend that you use `smit install_all`, which will install the prerequisites and commit the updates.

Using installp

About this task

To install a PTF by using the `installp` command and write the output to a log file, follow these steps:

Run the command:



`installp -aXgd` *`PTF_filesets_location`* `-e` *`logfile fileset_names`*

-ag
Specifies that all the latest installable filesets available in the *PTF_filesets_location* directory are applied.

-p
Performs a preview of the installation process by running the preinstallation checks. See "Previewing the installation and license agreements" on page 6.

-X
Attempts to expand the file system at the default location if there is insufficient space to complete the install.

-d *`PTF_filesets_location`*
Specifies the directory where the filesets are located. This path might also be a mounted CD-ROM drive.

-e *`logfile`*
Specifies the name and location of the installation log file.

`fileset_names`
Specifies a list of names of the filesets in *PTF_filesets_location* that you want to install.

Note: The keyword *all* can be used to indicate that all filesets must be attempted during installation.

The following example installs all available PTF filesets in the `/compiler/update/` source directory to the default locations, as specified in "Filesets" on page 1, and writes an installation log file to `/tmp/install.log`.

Example:

```
installp -aXgd /compiler/update/ -e /tmp/install.log all
```

Updating an advanced installation using vacndi

If the XL C compiler has been installed to a non-default directory, you must use `vacndi` to install PTFs.

About this task

Note: The `vacndi` tool enforces that you install the PTF vac.ndi fileset before using the tool to install the PTF to a non-default location.

To update a non-default installation, do the following:

Procedure

1. Begin by installing the vac.ndi fileset included in the PTF. This fileset must be installed first, so that the `vacndi` script inside it can be used to correctly install the remaining filesets.

   To determine the version number of the vac.ndi fileset currently installed on your system, run the command:

   ```
   lslpp -l vac.ndi
   ```

   To install a new version of the vac.ndi fileset, run the command:

   ```
   installp -aYgd ptf_images_location -e logfile vac.ndi
   ```

2. Create a text file listing the PTF filesets you want to install. This text file must contain the name of a single PTF file on each line. You can use the **ls** command to do this: `ls source_path > ptf_names_file`.

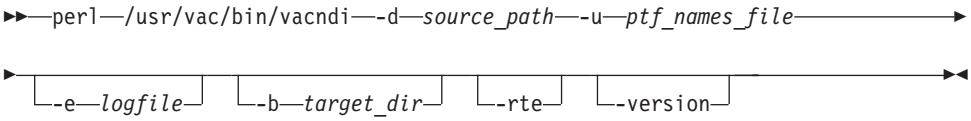
   For example:

   ```
   ls /compiler/update/*.bff > /home/user/update.list
   ```

   `update.list` contains all the updated fileset names, one for each line. For example:

   ```
   vac.C.12.01.0000.0001.bff
   ```

3. Use the perl command to run the `vacndi` script:


   ```
   perl /usr/vac/bin/vacndi -d source_path -u ptf_names_file [-e logfile]
   -b [target_dir]
   ```

   where:

   -d *source_path*
   Specifies the directory where the filesets are located. This path might also be a mounted CD-ROM drive.

   -u *ptf_names_file*
   Specifies the text file containing the names of PTF filesets you want to install.

   -e *logfile*
   Specifies the name and location of the installation log file. By default, the installation log file vacndi.log will be stored in your working directory.

   -b *target_dir*
   Specifies the location where the filesets should be copied and expanded. By default, the files are copied to the vacndi directory in your working directory. If the directory exists already, an error message is produced and the installation stops.

   -rte
   Specifies that only the runtime component is installed.

   -version
   Displays the version of the non-default installation tool.

Note: The installation script creates a compressed TAR backup of the compiler files that exist before the PTF update. If a problem is encountered with the installation of the PTF update or with the PTF update itself, you can restore from this backup file `target_dir.tar.Z` to uninstall the PTF update.

The following example uses the vacndi Perl script to install all available PTF filesets in the `/compiler/update/` source directory to the non-default location, `/compiler/xlc/` and writes the installation log file to `/tmp/xlc.ndi.inst.log`.

Example:

```
perl /usr/vac/bin/vacndi -d /compiler/update/ -u /compiler/update/ptflist
-b /compiler/xlc/ -e /tmp/xlc.ndi.inst.log
```

Product version migration

If you have a previous version of the compiler, you can replace it with the current version. You can also maintain multiple versions of the compiler on your system.

Before installing IBM XL C for AIX, V12.1, any previously installed versions of the compiler should be uninstalled. If you intend to maintain multiple versions of XL C on your system, refer to Chapter 3, "Advanced installation," on page 15, otherwise see Chapter 6, "Uninstalling XL C," on page 35.

If you are upgrading to XL C V12.1 using a basic installation, any previous versions of the filesets already installed on your system are automatically detected and migrated to the current version.

Once IBM XL C for AIX, V12.1 is installed, older compiler versions *cannot* be installed unless V12.1 is uninstalled first.

Operating system migration

If you have the compiler installed on AIX 5.3 or AIX 6.1, you might want to migrate the operating system to AIX 7.1.

The compiler filesets are not OS specific and do not require migration if XL C V12.1 is installed on AIX 5.3 and the operating system is migrated to AIX 6.1 or higher. If an OS specific fileset is required in the future for a higher level of AIX, it will be identified in a Tech Note tied to the PTF that adds support for that version of AIX.

Chapter 5. After installing XL C

There are optional setup and verification procedures that you can use after you have installed the compiler. You might also want to use the product documentation.

These are documented in the following sections:

Checking compiler and fileset information

You can check the versions of the compiler and individual filesets installed by using the **lslpp** command, the **-qversion** compiler option, or by viewing the `.vrmf_history` file.

Using lslpp

You can view information on the installed filesets by using the **lslpp** command:

```
lslpp option_string
```

where *option_string* can be:

-L *fileset*
: Displays the version and state of the fileset specified

-f *fileset*
: Displays the files and symbolic links associated with the fileset specified

-w *filename*
: Indicates which fileset the specified file (*filename*) belongs to

Note: lslpp detects only filesets that have been installed using the SMIT or installp methods.

Using -qversion

To get more details about the version, release, and PTF levels of the compiler, you can use the **-qversion** compiler option. For more information, see the *XL C Compiler Reference*.

Viewing the .vrmf_history file

If you have installed XL C to a non-default location, you can view the version information for each fileset installed by viewing the `.vrmf_history` text file which is installed in the compiler's main directory.

Note: You must not edit this file because it might cause unexpected behavior of the `vacndi` script.

Enabling utilization tracking

Utilization tracking can be used to detect whether the compiler usage exceeds your entitlement based on the number of Concurrent User licenses you have purchased.

Utilization tracking is disabled by default. You can enable it by modifying the entry **-qnoenabletracking** to **-qenabletracking** in the utilization tracking configuration file, urtxlc1201aix.cfg. Various other aspects of utilization tracking can also be specified in this file. In each installation of the compiler, there is a symlink named `urt_client.cfg` that points to the utilization tracking configuration file. You might need to change the symlink if you use a copy of the utilization tracking configuration file in a different location or if you use multiple copies of the file. For more information about utilization tracking, see Configuring utilization tracking in the *XL C Compiler Reference*.

Enabling IBM License Metric Tool (ILMT) and Tivoli Asset Discovery for Distributed (TADd)

IBM License Metric Tool (ILMT) and Tivoli® Asset Discovery for Distributed (TADd) can help you manage software usage metering and license allocation services on supported systems. In general, ILMT and TADd recognize and monitor the products that are installed and in use on your system.

ILMT and TADd are enabled for IBM XL C for AIX, V12.1 for inventory support only, which means that ILMT and TADd are able to detect product installation of XL C, but not its usage.

Note: ILMT and TADd are not a part of the XL C compiler offering, and must be purchased and installed separately.

Once installed and activated, ILMT and TADd scan your system for product inventory signatures that indicate whether a given product is installed on your system. ILMT and TADd also identify the version, release, and modification levels of the product. Inventory signature files are not updated after PTF installation.

If XL C is installed in the default location, the signature files are in the `/usr/vac/` directory. Otherwise, if XL C is installed in a non-default location, the signature files are in the */target_dir*`/usr/vac/` directory, where *target_dir* is the target directory for installation specified by the **-b** option of the non-default installation script.

For more information, see IBM License Metric Tool and Tivoli Asset Discovery for Distributed.

Configuring compiler defaults

You can change some of the default settings of the compiler for your specific needs.

If you have installed XL C onto a system running AIX 5.3, the configuration file is /etc/vac.cfg.53, and if you have installed the compiler onto a system running AIX 6.1, the configuration file is /etc/vac.cfg.61. The GCC options configuration file is `/etc/gxlc.cfg`.

For more information about how to change the XL C compiler's settings, see Customizing the configuration file and Configuring the gxlc option mapping in the *XL C Compiler Reference*.

Setting environment variables

Before using the compiler, verify that the environment variables LANG, MANPATH, NLSPATH, and PATH are set.

You can use the **echo** command to determine the current setting of these variables:

```
echo "\n $LANG \n\n $NLSPATH \n\n $MANPATH \n\n $PATH"
```

LANG
: Specifies the national language for message and help files. See "National language support" on page 4.

  The LANG environment variable can be set to any of the locales provided on the system. See the description of locales in *AIX General Programming Concepts* for more information.

  The national language code for United States English is **en_US**. If the appropriate message catalogs have been installed on your system, any other valid national language code can be substituted for **en_US**.

MANPATH
: Optionally specifies the directory search path for finding man pages. MANPATH must contain `/usr/vac/man/$LANG` before the default man path.

NLSPATH
: Specifies the path name of the message and help files.

PATH
: Specifies the directory search path for the executable files of the compiler.

Note: The LANG and NLSPATH environment variables are initialized when the operating system is installed, and might differ from the ones you want to use.

Enabling the XL C error messages

If your system uses the en_US locale and encoding, the compiler message catalogs are automatically configured to display correctly, whether you used the basic or advanced method of installation and configuration. However, if your system uses any other supported locale, you must set the NLSPATH environment variable so that the compiler and runtime functions can find the appropriate message catalogs following installation.

For example, to specify the Japanese locale with the IBM_eucJP code page, use the following commands:

```
LANG=ja_JP
NLSPATH=/usr/lib/nls/msg/%L/%N:/usr/lib/nls/msg/prime/%N
export LANG NLSPATH
```

Substitute any valid national language code for **ja_JP**, provided the associated message catalogs are installed.

See "National language support" on page 4 for a list of supported language locales.

Enabling the manual pages

Manual pages are provided for the compiler invocation commands and other compiler utilities.

Before you can view the manual pages, you must add the manual page directory to the MANPATH environment variable by entering the following on the command line:

```
export MANPATH=$MANPATH:/usr/vac/man/LANG
```

where *LANG* may be any one of:

- en_US
- EN_US
- ja_JP
- JA_JP
- Ja_JP
- zh_CN
- ZH_CN
- Zh_CN

For example, to set the language of the manual pages to English, run the commands:

```
MANPATH=$MANPATH:/usr/vac/man/en_US
export MANPATH
```

Alternatively, to set the language of the manual pages to one of the other supported languages, run the command:

```
MANPATH=$MANPATH:/usr/vac/man/$LANG
export MANPATH
```

where $LANG is set to any of the language locales listed above.

To test whether the manual pages have been successfully installed, open one or more of the manual pages, as described in "Viewing the manual pages" on page 33.

Creating shortcuts to the XL C invocation commands

The compiler invocation commands are not automatically installed in /usr/bin/. To invoke the compiler without having to specify the full path, create a shortcut to the invocation commands.

To create a shortcut to the XL C invocation commands, do *one* of the following:

Either

- Create symbolic links for the specific drivers from /usr/vac/bin/ to /usr/bin/.

  Example:

  ```
  ln -s /usr/vac/bin/xlc /usr/bin/xlc
  ```

or

- Add /usr/vac/bin/ to your PATH environment variable.

  Example:

  ```
  PATH=$PATH:/usr/vac/bin/
  export PATH
  ```

Note: If XL C resides in a non-default location, the invocation commands are installed to /*target_dir*/usr/vac/bin where *target_dir* is the target directory for installation specified by the **-b** option of the non-default installation script. You can create a shortcut to the invocation commands of a non-default installation by adding the location of the compiler invocations, *target_dir*/usr/vac/bin/, to the PATH environment variable.

For example, if you installed XL C to the /compiler/xlc directory, you would run the commands:

```
PATH=$PATH:/compiler/xlc/usr/vac/bin/
export PATH
```

Accessing the local documentation

Help using IBM XL C for AIX, V12.1 is available in both HTML and PDF formats. Manual pages for the compiler invocation commands and other command utilities are also included.

Viewing the HTML documentation

The product documentation is available in a fully searchable HTML-based information center.

The information center is viewable on the web at http://publib.boulder.ibm.com/infocenter/comphelp/v121v141/index.jsp. You can also download a version of this information center containing the product documentation for IBM XL C for AIX, V12.1, IBM XL C/C++ for AIX, V12.1 and IBM XL Fortran for AIX, V14.1. This information center can be installed on any AIX 5.3, AIX 6.1 ,or AIX 7.1 system running in 64-bit mode. It can be accessed with a web browser from the machine on which it is installed or from other computers on the same network.

Notes:

- You must have root access to launch and shut down the information center.
- Before you run the information center, you must ensure the machine is running in the 64–bit mode. It must display 64 when you run the following command:

  ```
  bootinfo -K
  ```

- The compiler installation does not remove the information center installed from previous versions of the product. To remove the previously installed information center, run the following command:

  ```
  installp -u xlhelp.\*
  ```

- The compressed information center package file is approximately 160 MB, and 220 MB when it is uncompressed.

Follow these steps to download the help system:

1. Go to the XL C compiler library page at http://www.ibm.com/software/awdtools/xlc/aix/library/. Download the information center package file `XLHelp-v121-v141-AIX-64bit.tar.Z` from the download link on this page.
2. Extract the package on your machine. Run the following command to extract the files to a new directory at `./xlhelp`:

   ```
   zcat XLHelp-v121-v141-AIX64.tar.Z | tar -xf -
   ```

3. Enter the new directory.

   ```
   cd ./xlhelp
   ```

4. You can review the licence information in the following two files:
   - LicenseAgreement.pdf
   - LicenseInformation.pdf
5. Start the information center by running the following command:

   ```
   ./xlhelp &
   ```

6. You can access the information center either locally or remotely. You need to have a web browser installed on your machine to view the information center.
   - To access the information center locally, use this URL:

     `http://`*`localhost`*`:1214/help/index.jsp`
   - To access the information center remotely, use this URL:

     `http://`*`machine_name`*`:1214/help/index.jsp`

     where *machine_name* is the name of the computer on which the Eclipse server has been launched.
7. To stop the information center, run this command:

   ```
   ./xlhelp_end
   ```

8. To remove the information center, you must first stop the information center, and remove the extracted directory `.xlhelp` using the following command:

   ```
   rm -r ./xlhelp
   ```

Note: The downloaded information center does not contain the documentation for IBM Debugger for AIX, V12.1. You can access the debugger documentation, which consists of HTML online help and additional HTML documents, from its graphical user interface.

Viewing the PDF documentation

PDF versions of the XL C product manuals are available in the `/doc/`*`LANG`*`/pdf/` directory of the installation media (either product CD or electronic package).

After default installation, the PDF documentation can be found in the `/usr/vac/doc/`*`LANG`*`/pdf/` directory. For non-default installations, the PDF documentation is located in the `/`*`target_dir`*`/usr/vac/doc/`*`LANG`*`/pdf/` directory, where *target_dir* is the target directory for installation specified by the **-b** option of the non-default installation script.

LANG can be en_US, ja_JP or zh_CN.

See “Related information” on page ix for a list of IBM XL C publications and descriptions of the PDF files available with the product.

Viewing the manual pages

Man pages are included for all compiler invocation commands and utilities.

For instructions about how to enable the man pages for viewing, see “Enabling the manual pages” on page 30.

To invoke a manual page, run the command:

```
man command
```

where *command* is any XL C invocation or utility command.

Example:

```
man xlc
```

Testing the installation

To test the product installation and the critical search paths, try building, compiling, and running a simple program.

For example:

1. Create the following C program:

   [hello.c]

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

2. Compile the test program. For example:

   ```
   xlc hello.c -o hello
   ```

   Note: To use shortcuts to the compiler invocation commands, see “Creating shortcuts to the XL C invocation commands” on page 31. Otherwise, include the full path to the commands to compile the program.

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 is zero.

Chapter 6. Uninstalling XL C

This section outlines the steps required to uninstall both basic and advanced installations of XL C.

Note:

1. You must have root user access to uninstall XL C.
2. Some filesets might not be uninstalled if they are required by other installed products. See "Filesets" on page 1 for details about filesets included with IBM XL C for AIX, V12.1.
3. Because automatically uninstalling dependent packages might cause problems, it is recommended that you preview uninstallation to ensure that all dependent filesets are no longer required.

Uninstalling a basic installation

To uninstall the compiler from the default location, you can use either **SMIT** or **installp** to remove the compiler filesets.

If you attempt to remove a fileset that is required by another installed fileset, the selected fileset will not be removed unless its dependents are also being removed.

Using SMIT

The following steps illustrate how to uninstall IBM XL C for AIX, V12.1 using the SMIT interface:

- Run the command:

  ```
  smit remove
  ```

- In the `SOFTWARE name` field, enter the fileset names (wildcards accepted) separated by a space. If you have XL Fortran installed in its default location, you must not remove any filesets which are shared between the compilers. In this case, enter the following name in the field:

  ```
  vac.*
  ```

  Otherwise, enter the following name in the field:

  ```
  vac.* memdbg.* xlhelp.* xlmass.* xlsmp.*
  ```

  To uninstall IBM Debugger for AIX, V12.1, you also need to enter the following name in the field:

  ```
  ibmdebugger
  ```

  In the `REMOVE dependent software?` field, select **yes**.

  Note: To preview the uninstallation, set the `PREVIEW only?` value to **yes**.

Using installp

To uninstall IBM XL C for AIX, V12.1 using the **installp** command, enter the following command on the command line:

```
installp -ug filesets
```

where *filesets* is the list of filesets (wildcards accepted) to uninstall.

If you have XL Fortran installed in its default location, you must not remove any filesets which are shared between the compilers. In this case, *filesets* should be:

```
vac.\*
```

Otherwise, *filesets* are:

```
vac.\* memdbg.\* xlmass.\* xlsmp.\*
```

To uninstall IBM Debugger for AIX, V12.1, enter the following command:

```
installp -ug ibmdebugger
```

Uninstalling an advanced installation

If the compiler is installed to a non-default location, you have to remove the filesets yourself.

About this task

To uninstall the XL C compiler installed to a non-default location, delete the compiler directory.

To determine which version of each fileset is installed, see "Checking compiler and fileset information" on page 27.

To delete the compiler directory, enter the following command on the command line:

```
rm -rf target_dir
```

where *target_dir* is the target directory for installation specified by the **-b** option of the non-default installation script.

Uninstalling versions of the MASS components downloaded from the web

If you have previously installed the MASS libraries on the system on which you are installing the IBM XL C for AIX, V12.1, it is recommended that you uninstall the libraries by removing the directory in which they have been installed.

About this task

To remove the MASS directory, enter:

```
rm -rf mass_directory
```

where *mass_directory* is the location of the MASS libraries. If you have created symbolic links in the `/usr/lib/` directory to the libraries in the MASS directory, you must delete them as well. The compiler installation automatically handles the creation of the necessary symbolic links.

Uninstalling the electronic license agreement

The license agreement text file remains on the system after the license fileset has been uninstalled, so that you do not have to accept the product license again if you reinstall XL C.

About this task

To remove the license agreement text file, run the following commands:

```
ODMDIR=/usr/lib/objrepos; odmdelete -o lag -q "fileset=vac.licAgreement"
rm -rf /usr/swlag/vac/121
```

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 websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites 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
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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 1998, 2012. All rights reserved.

Trademarks and service marks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

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

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Index

Special characters

A

B

C

D

E

F

H

I

L

M

N

O

P

S

T

U

Product Number: 5765-J01; 5725-C71

Printed in USA

GC14-7324-00