

**IBM XL C/C++ Advanced Edition V8.0 for
Linux**



インストール・ガイド

**IBM XL C/C++ Advanced Edition V8.0 for
Linux**



インストール・ガイド

お願い

本書および本書で紹介する製品をご使用になる前に、41 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM XL C/C++ Advanced Edition V8.0 for Linux (プログラム番号 5724-M16) および、新しい版で明記されない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。製品のレベルに正しい版を使用していることを確認してください。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： GC09-8017-01
IBM XL C/C++ Advanced Edition V8.0 for Linux
Installation Guide

発 行： 日本アイ・ピー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2006.7

この文書では、平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、
平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 2003, 2006. All rights reserved.

© Copyright IBM Japan 2006

目次

本書について	v
本書の対象読者	v
本書の読み方	v
本書の構成	vi
本書の規則	vii
関連情報	ix
技術サポート	xi
将来の更新および文書正誤表	xi

第 1 章 XL C/C++ のインストール前の確認事項

インストール・イメージおよびパッケージ	1
インストール・パッケージ	1
各国語サポート	2
必要な作業の判別	3
基本インストールの作業	3
上級者向けインストールの作業	4
システム前提条件	6
使用可能なハード・ディスク・スペースの容量の確認	8
必要な GNU、Perl、および Java パッケージがインストールされていることの確認	9

第 2 章 基本インストール

新規インストールで <code>xlc_install</code> ユーティリティを実行する	11
<code>xlc_install</code> オプション	13

第 3 章 上級者向けインストール

XL C/C++ V8.0 をインストールして前のバージョンと共存させる	15
前のバージョンと同じロケーションに共存させる	
XL C/C++ V8.0 のインストール	16
XL C/C++ をデフォルト・ロケーション以外の場所にインストールする	17
すべてのパッケージを単一のデフォルト・ロケーション以外の場所にインストールする	17
パッケージを複数のデフォルト・ロケーション以外の場所にインストールする	18

第 4 章 更新のインストール

<code>xlc_install</code> ユーティリティを実行して、基本インストールを更新する	21
---	----

第 5 章 XL C/C++ の構成 (上級者向け)

<code>new_install</code> ユーティリティを実行する	26
<code>vac_configure</code> ユーティリティを直接実行する	26
<code>vac_configure</code> オプション	26

第 6 章 XL C/C++ のインストール後の処置

インストール済みパッケージの照会	29
インストールのテスト	30
基本例: "Hello World" の作成および実行	30
マニュアル・ページの使用可能化	31
エラー・メッセージの使用可能化	31
呼び出しコマンド用の環境のセットアップ	32
コンパイラ呼び出しへのパスを組み込むように PATH 環境変数を設定する	33
コンパイラ呼び出しへのシンボリック・リンクの作成	33

第 7 章 XL C/C++ のアンインストール

例 (SLES10、Y-HPC、YDL): XL C/C++ V8.0 for Linux のアンインストール	35
---	----

第 8 章 インストールおよび構成のトラブルシューティング

エラー・メッセージおよび推奨処置	37
指定されたディレクトリ <code>rpmlocation_path</code> が存在しない	37
<code>rpmlocation_path</code> に...が含まれていない	37
32 ビットまたは 64 ビット GCC (RHEL4U2、Y-HPC、YDL) の場所を判別できなかった	38

特記事項

プログラミング・インターフェース情報	42
商標	43
業界標準	43

索引

本書について

本書には、IBM® XL C/C++ Advanced Edition V8.0 for Linux® のインストールについての基本的情報があります。この製品をインストールする前に、本書をよくお読みください。CD の README ファイルには製品の最新情報が含まれていますので、このファイルも必ずお読みください。製品のインストール後は、README ファイルは `installation_path/vac/8.0/` ディレクトリーにあります。この場合、`installation_path` は、システム上のコンパイラーの場所です。コンパイラーをデフォルト・ロケーションにインストールする場合は、`installation_path` は `/opt/ibmcomp/` です。

本書の対象読者

本書は、IBM XL C/C++ Advanced Edition V8.0 for Linux をインストールする職責を持つ方を対象としています。

本書は、基本インストール方式を使用される大部分のユーザーの方の要望に答えるもので、インストール・プロセスでのガイダンスを示しています。基本例は、基本インストールの手順を可能な限り反映するようになっています。

本書は、XL C/C++ の複数のバージョンを単一のシステムで維持する場合など、さまざまな目的に合わせてカスタマイズされたインストールを実行したいユーザーの要望にも対応しています。こうしたユーザーは、コンパイラー・インストールの経験が豊富で、システムにインストール済みの全コンパイラー製品の全バージョンのファイル構造に精通しておられます。本書では、こうしたユーザーを上級者と呼んでいます。上級者に必要な追加情報には、『上級者向け』というラベルが付いています。

本書の読み方

本書では、考えられる 3 つの主なインストール・シナリオでの手順を示します。

『基本』インストール

このシナリオでは、単一バージョンの XL C/C++ をデフォルト・ロケーションにインストールできます。大部分のユーザーに適用でき、製品のインストールに推奨できる方法です。基本インストールを実行する場合に従う必要があるステップの概要については、3 ページの『基本インストールの作業』を参照してください。ここには、本書の関係する手順に対するリンクがあります。

『上級者向け』インストール

このシナリオでは、単一のシステムで複数バージョンの XL C/C++ を維持するか、製品をデフォルト・ロケーション以外の場所にインストールできます。このシナリオは、特殊なニーズを持つ上級者向けであり、多数のユーザーに推奨できるものではありません。上級者向けインストールを実行する場

合に従う必要があるステップの概要については、4 ページの『上級者向けインストールの作業』を参照してください。ここには、本書の関係する手順に対するリンクがあります。

『更新』 インストール

このシナリオは、既存の XL C/C++ V8.0 基本または上級者向けインストールの更新パッケージを入手したユーザーに適用されます。更新を実行する場合に従う必要があるステップの概要については、更新が必要なインストールのタイプに応じて 3 ページの『基本インストールの作業』または 4 ページの『上級者向けインストールの作業』を参照してください。

本書での取り扱い範囲外である特殊なインストール・シナリオについては、以下のサイトで Technotes を参照してください。

<http://www.ibm.com/software/awdtools/xlcpp/support>。

本書の構成

本書は、XL C/C++ のインストールに関して、プリインストール、インストール、ポストインストール、およびトラブルシューティングの各段階を反映するように編成されています。

表 1. XL C/C++ のインストールの段階

段階	章	対象ユーザー
プリインストール	1 ページの『第 1 章 XL C/C++ のインストール前の確認事項』	すべてのユーザー
インストール	11 ページの『第 2 章 基本インストール』	次のようなユーザー • 最も単純で、直接的なインストール・プロセスを使用したい • 特別な要件 (コンパイラーの複数バージョンの使用など) がない
	15 ページの『第 3 章 上級者向けインストール』	次のようなユーザー • コンパイラーをデフォルト・ロケーション以外の場所にインストールしたい。 • コンパイラーの複数バージョンを同じシステムにインストールしたい
	21 ページの『第 4 章 更新のインストール』	XL C/C++ V8.0 を次のフィックス・レベルに更新したいユーザー
ポストインストール	25 ページの『第 5 章 XL C/C++ の構成 (上級者向け)』	次のようなユーザー • 上級者用の、デフォルト以外の方法を使用してコンパイラーをインストールまたは更新したい • 前にデフォルト・ロケーション以外の場所にインストールされているコンポーネントを更新する必要がある
	29 ページの『第 6 章 XL C/C++ のインストール後の処置』	すべてのユーザー

表 1. XL C/C++ のインストールの段階 (続き)

段階	章	対象ユーザー
製品の除去	35 ページの『第 7 章 XL C/C++ のアンインストール』	システムから XL C/C++ コンパイラーを除去する必要があるユーザー
トラブルシューティング	37 ページの『第 8 章 インストールおよび構成のトラブルシューティング』	XL C/C++ のインストールまたは構成時に、エラー・メッセージまたは予期しない結果への応答方法を知る必要があるユーザー

本書の規則

書体の規則

以下の表では、本書で使用されている書体の規則を説明します。

表 2. 書体の規則

書体	書体が示す対象	例
太字	コマンド、実行可能ファイル名、コンパイラー・オプション、およびプラグマ・ディレクティブ	-O3 を指定すると、コンパイラーは -qhot=level=0 とみなします。 -O3 によるすべての HOT 最適化を避けるには、 -qnohot を指定する必要があります。
イタリック	ユーザーが実際の名前または値を指定するパラメーターまたは変数。イタリックは、新規用語を紹介するためにも使用されます。	要求した <i>size</i> を超える値を戻す場合は、 <i>size</i> パラメーターを必ず更新してください。
モノスペース	プログラミング・キーワードとライブラリー関数、コンパイラー組み込み関数、プログラム・コードの例、コマンド・ストリング、またはユーザー定義の名前	switch 文の 1 つまたは 2 つのケースが他のケースに比べて一般的に実行頻度が高い場合、そのケースを switch 文の前に個別に処理することによって取り出します。

構文図

本書では、構文図は XL C/C++ 構文を示します。この節では、構文図の解釈方法と使用法を示します。

- 構文図は線の経路に沿って、左から右へ、上から下へ読みます。

▶— 記号は、コマンド、ディレクティブ、または文の始まりを示します。

→ 記号は、コマンド、ディレクティブ、または文の構文が次の行に続くことを示します。

▶— 記号は、コマンド、ディレクティブ、または文が前の行からの続きであることを示します。

→▶ 記号は、コマンド、ディレクティブ、または文の終わりを示します。

完全なコマンド、ディレクティブ、または文以外の構文単位の断片図は、|— 記号で始まり、—| 記号で終わります。

- 必須項目は、次のように水平線（メインパス）上に表示されます。



- オプション項目は、メインパスの下側に表示されます。



- 複数の項目から選択できる場合は、縦に並べて表示されます。

複数の項目から 1 つを選択しなければならない 場合は、縦の並びの中のいずれか 1 つの項目がメインパス上に表示されます。



複数の項目からの選択がオプションの場合は、縦の並び全体がメインパスの下側に表示されます。



- 主線上を左方へ戻る矢印（繰り返し矢印）は、縦に並べて指定されている項目から複数の項目を選択できること、または選択項目を繰り返して指定できることを示しています。ブランク以外の分離文字も示されます。



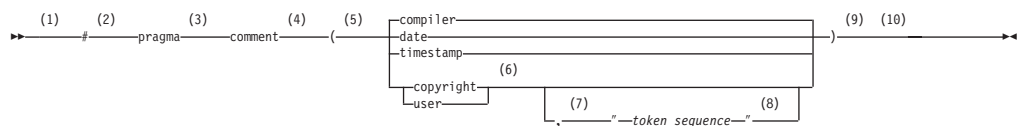
- デフォルトの項目は、メインパスの上側に表示されます。



- キーワードは非イタリック体の文字で示され、そのとおり正確に入力する必要があります。
- 変数はイタリック体の小文字で示されます。変数は、ユーザー指定の名前や値を表します。
- 句読記号、括弧、算術演算子、またはその他の記号が示されている場合は、それらを構文の一部として入力する必要があります。

構文図の例

以下の構文図の例は、**#pragma comment** ディレクティブの構文を示しています。



注:

- これは構文図の始まりを示します。
- 記号 # が先頭になければなりません。

- 3 キーワード `pragma` が # 記号の後に続く必要があります。
- 4 プラグマ `comment` の名前がキーワード `pragma` の後に続く必要があります。
- 5 左括弧が必要です。
- 6 コメント・タイプは、次に示すタイプの 1 つのみとして入力する必要があります。`compiler`、`date`、`timestamp`、`copyright`、または `user`。
- 7 コメント・タイプ `copyright` または `user` とオプションの文字ストリングの間には、コンマが必要です。
- 8 コンマの後に文字ストリングを続けなければなりません。文字ストリングは二重引用符で囲む必要があります。
- 9 右括弧が必要です。
- 10 これは構文図の終わりを示します。

以下の **#pragma comment** ディレクティブの例は、上記の図と構文的に一致しており、正しく入力された例です。

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

例および基本例

本書の例は、『例』または『基本例』というラベルが付いています。基本例 は、基本インストール中に実行される手順をほとんどそのまま、または変更しないで示すことを意図したものです。

関連情報

IBM XL C/C++ の資料

XL C/C++ では、製品資料が以下のフォーマットで提供されます。

- README ファイル

README ファイルには、製品資料への変更と修正を含む最新の情報が収められています。README ファイルは、デフォルトでは `/opt/ibmcmp/vacpp/8.0/` ディレクトリにあり、またインストール CD のルート・ディレクトリにあります。

- インストール可能なマニュアル・ページ

マニュアル・ページは、製品と共に提供されるコンパイラー呼び出しおよびすべてのコマンド行ユーティリティに対して提供されています。マニュアル・ページのインストールおよびアクセス手順は、本書 にあります。

- インフォメーション・センター

XL C/C++ HTML 文書のインフォメーション・センターは 製品とともに出荷されます。HTML 文書は、デフォルトでは `/opt/ibmcmp/vacpp/8.0/doc/language/html/` ディレクトリ (ここで、*language* は `en_US`、`zh_CN`、または `ja_JP` のいずれか) にあります。検索可能 HTML のインフォメーション・センターは、以下の Web 上でも表示可能です。

<http://publib.boulder.ibm.com/infocenter/lnxpcomp/v8v101/index.jsp>

- PDF 文書

PDF 文書は、デフォルトでは /opt/ibmcomp/vacpp/8.0/doc/language/pdf/ ディレクトリー (ここで、*language* は en_US、zh_CN、または ja_JP のいずれか) にあります。PDF は、以下の Web サイトでも入手できます。

<http://www.ibm.com/software/awdtools/xlcpp/library>

この資料に加えて、以下のファイルが XL C/C++ 製品マニュアルのセットを構成しています。

表 3. XL C/C++ PDF ファイル

文書タイトル	PDF ファイル名	説明
IBM XL C/C++ 入門 <i>Advanced Edition V8.0 for Linux, SD88-6730-00</i>	getstart.pdf	XL C/C++ 製品の概要説明のほか、ユーザー環境のセットアップと構成、プログラムのコンパイルとリンク、およびコンパイル・エラーのトラブルシューティングに関する情報が含まれます。
IBM XL C/C++ Advanced Edition V8.0 for Linux <i>Compiler Reference, SD88-6729-00</i>	compiler.pdf	さまざまなコンパイラー・オプション、プラグマ、マクロ、環境変数、および組み込み関数 (並列処理に使用されるものを含む) についての情報が含まれます。
IBM XL C/C++ Advanced Edition V8.0 for Linux <i>Language Reference, SC09-8016-00</i>	language.pdf	IBM がサポートする C および C++ プログラミング言語についての情報のほか、ポータビリティおよび非機密標準への準拠に対応した言語拡張機能の情報が含まれます。
IBM XL C/C++ Advanced Edition V8.0 for Linux <i>Programming Guide, SC09-8014-00</i>	proguide.pdf	拡張プログラミングのトピック (アプリケーション・ポーティング、FORTRAN コードでの言語間呼び出し、ライブラリー開発、アプリケーション最適化および並列化など)、および XL C/C++ ハイパフォーマンス・ライブラリーの情報が含まれます。

これらの PDF ファイルは、Adobe Reader で表示でき、印刷できます。Adobe Reader がインストールされていない場合は、<http://www.adobe.com> からダウンロードできます。

XL C/C++ に関連した Redbooks、ホワイト・ペーパー、チュートリアル、およびその他の論文を含む追加のドキュメンテーションが以下の Web サイトから入手できます。

<http://www.ibm.com/software/awdtools/xlcpp/library>

その他の IBM 資料

- POWER 関係の資料に関する Engineering and Scientific Subroutine Library (ESSL) for Linux は、<http://publib.boulder.ibm.com/infocenter/clresctr/index.jsp> から入手できます。

技術サポート

追加の技術サポートを XL C/C++ Support ページから利用することができます。このページは、技術サポート FAQ、Technotes、および他のサポート文書を広範囲に選択するための検索機能を持つポータルを提供します。XL C/C++ Support ページは、次の Web サイトにあります。

<http://www.ibm.com/software/awdtools/xlcpp/support>

必要な情報を検索できない場合は、次のアドレスに E メールをお送りください。

compinfo@ca.ibm.com

XL C/C++ についての最新情報は、次のサイトで製品情報を参照してください。

<http://www.ibm.com/software/awdtools/xlcpp>

将来の更新および文書正誤表

この製品に対する将来のすべての更新の詳細および文書正誤表は、以下の URL にあります。

- <http://www.ibm.com/support/docview.wss?uid=swg21220307>

第 1 章 XL C/C++ のインストール前の確認事項

IBM® XL C/C++ Advanced Edition V8.0 for Linux® をインストールする前の確認事項

- 製品の README ファイルを参照して、直前の更新の有無を確認します。
- インストール可能コンパイラー・パッケージおよびインストール用のユーティリティー・プログラムが入っているインストール・イメージについて十分に理解します。
- 実行する必要がある作業を決定します。これは、インストール要件によって決まります。
- root ユーザーまたは管理者特権を持つユーザーになります。
- システム前提条件が満たされていること、および必要なソフトウェア・パッケージがすべてインストールされていることを確認します。

インストール・イメージおよびパッケージ

IBM XL C/C++ Advanced Edition V8.0 for Linux インストール・イメージは、インストール CD からか、IBM Web サイトからローカル・ドライブにダウンロードすれば入手できます。

イメージには、以下のものが含まれます。

- README、使用許諾契約書ファイル、および解説書
- RPM パッケージのセット。『インストール・パッケージ』を参照してください。
- 基本インストール用にコンパイラーをインストールし、構成するためのインストール・ツール (**xlc_install**)。3 ページの『基本インストールの作業』を参照してください。
- サポートされる各国語ごとのメッセージ・カタログ。2 ページの『各国語サポート』を参照してください。

インストール・パッケージ

2 ページの表 4 には、インストール・イメージと共に提供されるパッケージおよび基本インストール時にデフォルトでインストールされる場所がリストされています。(カスタムのデフォルト・ロケーション以外の場所にパッケージをインストールする場合の規則については、18 ページの表 14 を参照してください。)

Linux **rpm** ユーティリティーを使用して、パッケージを検討することができます。例えば、パッケージ情報およびそのファイル・リストを表示するには、次の **rpmQUERY** コマンドを発行します。

```
# rpm -qip package_name
```

表 4. XL C/C++ for Linux パッケージおよびデフォルトのインストール・ロケーション

パッケージ名	パッケージの説明	デフォルトのインストール・ロケーション
xlsmp.msg.rte	XL SMP ランタイム・メッセージ	/opt/ibmcmp/msg/
xlsmp.rte	XL SMP ランタイム動的ライブラリー	/opt/ibmcmp/lib/ /opt/ibmcmp/lib64/
xlsmp.lib	XL SMP ランタイム静的ライブラリー	/opt/ibmcmp/xlsmp/1.6
xlmass.lib	IBM Mathematical Acceleration Subsystem (MASS) ライブラリー	/opt/ibmcmp/xlmass/4.3
vacpp.rte	XL C/C++ ランタイム環境	/opt/ibmcmp/lib/ /opt/ibmcmp/lib64/
vacpp.rte.lnk	XL C/C++ ランタイム環境リンク	/opt/ibmcmp/vacpp/8.0/
vac.lic	XL C/C++ ライセンス	/opt/ibmcmp/vac/8.0/
vac.lib	XL C コンパイラー・ライブラリー	/opt/ibmcmp/vac/8.0/
vac.cmp	XL C コンパイラー・ファイル	/opt/ibmcmp/vac/8.0/
vacpp.lib	XL C++ コンパイラー・ライブラリー	/opt/ibmcmp/vacpp/8.0/
vacpp.cmp	XL C++ コンパイラー・ファイル	/opt/ibmcmp/vacpp/8.0/
vacpp.samples	XL C/C++ サンプル	/opt/ibmcmp/vacpp/8.0/サンプル/
vacpp.help	XL C/C++ の資料	/opt/ibmcmp/vacpp/8.0/doc/ /opt/ibmcmp/vacpp/8.0/man/

各国語サポート

XL C/C++ メッセージは、以下の言語ロケールをサポートします。

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

英語 (en_US) が各国語のデフォルトです。以下のインストールでは、メッセージが異なる言語で表示されるように NLSPATH を設定することができます。 31 ページの『エラー・メッセージの使用可能化』を参照してください。

必要な作業の判別

以下の各節に記載されている表を使用すると、製品をインストールして構成する際に必要な情報を見つけることができます。

基本インストールの作業

以下の条件がすべて該当する場合に限り、『基本』(デフォルト) のインストール方法を使用するようにお勧めします。

- システムで製品の単一バージョンを維持する。この場合、XL Fortran Advanced Edition V10.1 for Linux がある場合も、ない場合もあります。
- 製品をデフォルト・ロケーション (/opt/ibmcomp/) にインストールする。

以上の条件でニーズが満たされる場合、基本インストールが最も簡単で、時間のかからない方法です。単一のインストール・ツールを使用するだけで、コンパイラーの前のバージョンのアンインストール、最新のバージョンのインストール、およびコンパイラーの構成を自動的行うことができるからです。

はじめて XL C/C++ V8.0 をインストールする場合は、表 5 の手順に従ってください。XL C/C++ V8.0 の更新をインストールする場合は、4 ページの表 6 の手順に従ってください。

表 5. 基本インストールのステップ: 新規またはアップグレードのインストール

作業	詳細についての参照先 . . .
root ユーザーまたは管理者特権を持つユーザーになります。	オペレーティング・システムと共に提供される資料
システム前提条件がすべて満たされていることを確認します。	6 ページの『システム前提条件』
デフォルト・パスを使用し、 xlc_install ツールを使用して、コンパイラーをインストールおよび構成します。	11 ページの『第 2 章 基本インストール』
コンパイラー・パッケージが正常にインストールされたことを確認して、インストールをテストします。	<ul style="list-style-type: none">• 29 ページの『インストール済みパッケージの照会』• 30 ページの『インストールのテスト』
コンパイラーのマニュアル・ページを使用可能にします。	31 ページの『マニュアル・ページの使用可能化』
システムのロケールまたはエンコードが英語 (en_US) でない場合は、コンパイラー・エラー・メッセージを使用可能にします。それ以外の場合は、このステップをスキップできます。	31 ページの『エラー・メッセージの使用可能化』

表 5. 基本インストールのステップ: 新規またはアップグレードのインストール (続き)

作業	詳細についての参照先 . . .
(オプション) インストール・プロセスでコンパイラ呼び出しコマンドへのシンボリック・リンクの作成を選択していない場合、絶対パスを指定しなくても呼び出しコマンドが見つかるように環境をセットアップします。それ以外の場合は、このステップをスキップできます。	32 ページの『呼び出しコマンド用の環境のセットアップ』

表 6. 基本インストールのステップ: 更新のインストール

作業	詳細についての参照先 . . .
root ユーザーまたは管理者特権を持つユーザーになります。	オペレーティング・システムと共に提供される資料
システム前提条件がすべて満たされていることを確認します。	6 ページの『システム前提条件』
xlc_install ツールを使用して更新パッケージをインストールします。	21 ページの『 xlc_install ユーティリティーを実行して、基本インストールを更新する』
コンパイラ・パッケージが正常にインストールされたことを確認して、インストールをテストします。	<ul style="list-style-type: none"> 29 ページの『インストール済みパッケージの照会』 30 ページの『インストールのテスト』
(オプション) 更新プロセスでコンパイラ呼び出しコマンドへのシンボリック・リンクの作成を選択していない場合、絶対パスを指定しなくても呼び出しコマンドが見つかるように環境をセットアップします。それ以外の場合は、このステップをスキップできます。	32 ページの『呼び出しコマンド用の環境のセットアップ』

上級者向けインストールの作業

以下の場合、インストールの『上級者向け』方法を使用する必要があります。

- 単一のシステムで同じ製品の複数バージョンを維持する。
- 製品をデフォルト・ロケーション以外の場所にインストールする。

これらいずれかの条件が該当する場合は、『上級者向け』のインストール方法を使用します。この方法の場合、コンパイラを個別にインストールして構成する必要があります。コンパイラの前のバージョンを手動でシステムからアンインストールしなければならない場合もあります。

はじめて XL C/C++ V8.0 をインストールする場合は、表 7 の手順に従ってください。XL C/C++ V8.0 の更新をインストールする場合は、5 ページの表 8 の手順に従ってください。

表 7. 上級者向けインストールのステップ: 新規またはアップグレードのインストール

作業	詳細についての参照先 . . .
root ユーザーまたは管理者特権を持つユーザーになります。	オペレーティング・システムと共に提供される資料

表 7. 上級者向けインストールのステップ: 新規またはアップグレードのインストール (続き)

作業	詳細についての参照先 . . .
システムに製品の複数バージョンを維持する必要がない場合、XL C/C++ for Linux の既存のバージョンを除去します。	35 ページの『第 7 章 XL C/C++ のアンインストール』
システム前提条件がすべて満たされていることを確認します。	6 ページの『システム前提条件』
上級者向けインストール方法の 1 つを使用して、コンパイラーをインストールします。	15 ページの『第 3 章 上級者向けインストール』
new_install または vac_configure ツールを使用してコンパイラーを構成します。	25 ページの『第 5 章 XL C/C++ の構成 (上級者向け)』
コンパイラー・パッケージが正常にインストールされたことを確認して、インストールをテストします。	<ul style="list-style-type: none"> • 29 ページの『インストール済みパッケージの照会』 • 30 ページの『インストールのテスト』
コンパイラーのマニュアル・ページを使用可能にします。	31 ページの『マニュアル・ページの使用可能化』
システムのロケールまたはエンコードが英語(en_US) でない場合は、コンパイラー・エラー・メッセージを使用可能にします。それ以外の場合は、このステップをスキップできます。	31 ページの『エラー・メッセージの使用可能化』
(オプション) 絶対パスを指定しなくても呼び出しコマンドが見つかるように環境をセットアップします。	32 ページの『呼び出しコマンド用の環境のセットアップ』

表 8. 上級者向けインストールのステップ: 更新のインストール

作業	詳細についての参照先 . . .
root ユーザーまたは管理者特権を持つユーザーになります。	オペレーティング・システムと共に提供される資料
システム前提条件がすべて満たされていることを確認します。	6 ページの『システム前提条件』
更新パッケージをデフォルト・ロケーション以外の場所にインストールします。	17 ページの『XL C/C++ をデフォルト・ロケーション以外の場所にインストールする』
コンパイラーを構成する	25 ページの『第 5 章 XL C/C++ の構成 (上級者向け)』
コンパイラー・パッケージが正常にインストールされたことを確認して、インストールをテストします。	<ul style="list-style-type: none"> • 29 ページの『インストール済みパッケージの照会』 • 30 ページの『インストールのテスト』
(オプション) 絶対パスを指定しなくても呼び出しコマンドが見つかるように環境をセットアップします。	32 ページの『呼び出しコマンド用の環境のセットアップ』

システム前提条件

以下は、IBM XL C/C++ Advanced Edition V8.0 for Linux をインストールする場合の要件です。

- **オペレーティング・システム:** サポートされる Linux ディストリビューション
 - Red Hat Enterprise Linux AS 4 Update 2 (RHEL4U2) for IBM POWER™
 - SUSE Linux Enterprise Server 9 Service Pack 2 (SLES9 SP2) for IBM POWER
 - SUSE Linux Enterprise Server 10 (SLES10) for IBM POWER
 - Terra Soft's Y-HPC v1.1 (Y-HPC) または Yellow Dog Linux v4.1 (YDL)
- **ハードウェア:** 以下のいずれのハードウェア・プラットフォームでも使用できます (ただしオペレーティング・システムの配布がサポートしている場合)。
 - IBM POWER テクノロジー・ベースのシステム
 - IBM BladeCenter® JS20
 - IBM System i™
 - IBM System p™
 - Apple Power Mac G5、Apple Xserver G5
- **ハード・ディスク・スペース所要量:**
 - 約 200 MB (製品パッケージ用)
 - 2 GB 以上 (ページング用)
 - 512 MB 以上 (一時ファイル用)

注: 高水準の最適化レベルの場合、ページングおよび一時ファイル用のスペースがさらに必要になることがあります。

使用可能なハード・ディスク・スペースが十分であることを確認するには、8 ページの『使用可能なハード・ディスク・スペースの容量の確認』の手順を参照してください。

- **必要なソフトウェア:**
 - GNU、Perl、および Java™ パッケージ (以下の表にリスト)。必要なパッケージがインストール済みであることを確認するには、9 ページの『必要な GNU、Perl、および Java パッケージがインストールされていることの確認』の手順を参照してください。

表 9. RHEL4U2 オペレーティング・システムに必要な GNU、Perl、および Java パッケージ

パッケージ名	バージョンの要件
gcc	3.4.4
gcc-c++	3.4.4
glibc	2.3.4
glibc-devel	2.3.4
libgcc	3.4.4
libstdc++	3.4.4
libstdc++-devel	3.4.4

表 9. *RHEL4U2* オペレーティング・システムに必要な *GNU*、*Perl*、および *Java* パッケージ (続き)

Perl	5.0 またはそれ以上 注: Perl V5.0 は、 <i>RHEL4U2</i> オペレーティング・システムと共に出荷され、自動的にインストールされます。
libgcj	3.4.4

表 10. *SLES9 SP2* オペレーティング・システムに必要な *GNU*、*Perl*、および *Java* パッケージ

パッケージ名	バージョンの要件
gcc	3.3.3
gcc-c++	3.3.3
gcc-64bit	9
glibc	2.3.3
glibc-64bit	9
glibc-devel-64bit	9
libgcc	3.3.3
libgcc-64bit	9
libstdc++	3.3.3
libstdc++-64bit	9
libstdc++-devel-64bit	9
Perl	5.0 またはそれ以上 注: Perl V5.0 は、 <i>SLES9 SP2</i> オペレーティング・システムと共に出荷され、自動的にインストールされます。
java2	1.3.1
java2-jre	1.3.1

表 11. *SLES10* オペレーティング・システムに必要な *GNU*、*Perl*、および *Java* パッケージ

パッケージ名	バージョンの要件
gcc	4.1.0
gcc-c++	4.1.0
glibc	2.4
glibc-64bit	2.4
glibc-devel-64bit	2.4
libgcc	4.1.0
libgcc-64bit	4.1.0
libstdc++	4.1.0
libstdc++-64bit	4.1.0
libstdc++-devel-64bit	4.1.0

表 11. SLES10 オペレーティング・システムに必要な GNU、Perl、および Java パッケージ (続き)

Perl	5.0 またはそれ以上 注: Perl V5.0 は、SLES10 オペレーティング・システムと共に出荷され、自動的にインストールされます。
libgcj	4.1.0

表 12. Y-HPC および YDL オペレーティング・システムに必要な GNU、Perl、および Java パッケージ

パッケージ名	バージョンの要件
gcc	3.4.4
gcc-ppc32	3.4.4
gcc-c++	3.4.4
glibc*	2.3.4
glibc-devel*	2.3.4
libgcc*	3.4.4
libstdc++*	3.4.4
libstdc++-devel*	3.4.4
Perl	5.0 またはそれ以上 注: Perl V5.0 は、Y-HPC または YDL オペレーティング・システムと共に出荷され、自動的にインストールされます。
libgcj	3.4.4

注: * 32 ビット・バージョンと 64 ビット・バージョンの両方が必要です。

• オプションのソフトウェア:

- K デスクトップ環境や GNOME などのグラフィカル・デスクトップ環境
- フレームが有効な HTML ブラウザー (ヘルプおよび他の Web ページにアクセスする場合)
- PDF ビューアー (PDF 資料にアクセスする場合)

使用可能なハード・ディスク・スペースの容量の確認

XL C/C++ には、約 200 MB のハード・ディスク・ストレージ・スペースが必要です。このハード・ディスク容量には、製品と共に出荷されるオプションのサンプルおよび資料も収容されます。

以下のコマンドを使用して、デフォルトのインストール・ロケーションで使用可能なスペースの量を判別できます (/opt/ibmcmp/)。

```
df -h /opt
```

コンパイラーをデフォルト・ロケーション以外の場所にインストールする計画がある場合は、以下のコマンドを使用できます。

```
df -h installation_path
```

ここで *installation_path* はデフォルト・ロケーション以外の場所です。

必要な GNU、Perl、および Java パッケージがインストールされていることの確認

XL C/C++ for Linux をインストールする前に、GNU、Perl、および Java パッケージの必要なバージョンがオペレーティング・システムと共にインストールされているかどうかを確認する必要があります。

サポートされる Linux 配布版ごとに必要なパッケージのリストについては、以下のいずれかを参照してください。

- 6 ページの表 9。RHEL4U2 オペレーティング・システムに必要な GNU、Perl、および Java パッケージ
- 7 ページの表 10。SLES9 SP2 オペレーティング・システムに必要な GNU、Perl、および Java パッケージ
- 7 ページの表 11。SLES10 オペレーティング・システムに必要な GNU、Perl、および Java パッケージ
- 8 ページの表 12。Y-HPC および YDL オペレーティング・システムに必要な GNU、Perl、および Java パッケージ

以下のコマンドを使用して、必要なパッケージの正しいバージョンがインストール済みかどうかを検査できます。

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

例: gcc-c++ のインストール済みバージョンの判別

gcc-c++ がインストール済みかどうかを調べるには、gcc-c++ パッケージについて次のように照会します。

```
rpm -qa | grep gcc-c++
```

gcc-c++ バージョン 4.1.0 がインストール済みの場合は、以下の出力と同様な結果が得られます。

```
gcc-c++-4.1.0-43.24
```

注: RHEL4U2、YDL、および Y-HPCでは、32 ビットと 64 ビットの両方の `glibc-devel` および `libstdc++-devel` パッケージが必要です。コンパイラをインストールする前に、これらのパッケージが使用可能であることを確認するには、38 ページの『32 ビットまたは 64 ビット GCC (RHEL4U2、Y-HPC、YDL) の場所を判別できなかった』の説明を参照してください。RHEL4U2、Y-HPC、および YDL 上のパッケージの名前は構造化されており、32 ビットのパッケージか 64 ビットのパッケージかに関係なく同じ名前であるため、この節の例を使用してこれらのパッケージがインストール済みかどうかを検査しないようにしてください。結果として、出力では 32 ビット、64 ビット、または両方のパッケージのいずれかがインストールされているかが示されません。

第 2 章 基本インストール

XL C/C++ には、基本インストールを段階的に進める対話式ユーティリティ (xlc_install) があります。xlc_install を使用すると、次のいずれでも行うことができます。

- XL C/C++ V8.0 for Linux をクリーン・システムにインストールする。
- XL C/C++ V8.0 for Linux を XL Fortran Advanced Edition V10.1 for Linux が既にインストールされているシステムにインストールする。

この場合、両方のコンパイラーが XL C/C++ for Linux で提供された XL SMP および IBM MASS ライブラリーを使用します。(XL Fortran V10.1 for Linux のインストールについては、「*XL Fortran Advanced Edition V10.1 for Linux* インストール・ガイド」を参照してください。)

- XL C/C++ V8.0 for Linux が既にインストール済みのシステムに更新をインストールする。

この場合、21 ページの『xlc_install ユーティリティを実行して、基本インストールを更新する』の手順に従ってください。

次の条件がすべて満たされた場合に限って、xlc_install ユーティリティを使用して XL C/C++ をインストールしてください。

- コンパイラーをデフォルト・ロケーション (/opt/ibmcomp/) にインストールする。
- 前にインストール済みの XL C/C++ コンポーネントを除去することに同意している。

以上の条件のうち 1 つでも満たされない場合、xlc_install ユーティリティを使用しないでください。この場合、15 ページの『第 3 章 上級者向けインストール』の手順に従ってください。

新規インストールで xlc_install ユーティリティを実行する

xlc_install ユーティリティは、インストール・イメージのルート・ディレクトリ内にあります。

注: xlc_install ユーティリティは Perl で書かれているため、ご使用のシステムに Perl がインストール済みであることを確認してから、このユーティリティを実行する必要があります。9 ページの『必要な GNU、Perl、および Java パッケージがインストールされていることの確認』を参照してください。

新規インストールで xlc_install ユーティリティを実行する場合は、以下のことが行われます。

- 前提のソフトウェア・パッケージがすべてチェックされます。
- 前にインストールされた XL C/C++ コンポーネントがアンインストールされます。

- すべてのコンパイラー・パッケージがデフォルト・ロケーションにインストールされます。
- **new_install** ユーティリティーが自動的に呼び出されます。このユーティリティーにより、ライセンス・ファイルがインストールされ、デフォルトの構成ファイルが生成されます。
- オプションでコンパイラー呼び出しコマンドへのシンボリック・リンクが `/usr/bin/` に作成されます。
- インストール・ログが `/tmp/` ディレクトリーに生成されます。

xlc_install ユーティリティーを実行して IBM XL C/C++ Advanced Edition V8.0 for Linux をインストールする方法

1. 製品 CD がシステムの `/cdrom` の場所にマウントされていることを前提として以下のコマンドを発行します。

```
# cd /cdrom
# ./xlc_install
```

xlc_install に指定できる追加の引数については、13 ページの『**xlc_install** オプション』を参照してください。

XL C/C++ for Linux の別のインスタンスがシステムで検出された場合、そのアンインストールを確認するプロンプトが表示されます。

2. アンインストールを進めることを確認します。コンパイラーの既存のインスタンスをアンインストールしないことを選択すると、インストール・プロセスが終了します。

XL SMP および IBM MASS パッケージのその他のバージョンがシステムで単独または XL Fortran Advanced Edition V10.1 for Linux のインストールの一部として検出された場合、そのアンインストールを確認するプロンプトが表示されます。

3. 既存の XL SMP および IBM MASS パッケージのアンインストールを進めることを確認します。前にインストール済みのコンポーネントをアンインストールしないことを選択すると、インストール・プロセスは終了します。

注: **xlc_install** は、これらのパッケージを既存の場所からアンインストールして、デフォルト・ロケーション (`/opt/ibmcomp/`) にインストールし直します。したがって、パッケージが前にデフォルト・ロケーション以外の場所に XL Fortran for Linux のインストールの一部としてインストール済みである場合は、**xlf_configure** を実行して、XL Fortran コンパイラーを再構成し、これらのパッケージのデフォルト・ロケーションを示す必要があります。手順については、「XL Fortran Advanced Edition V10.1 for Linux インストール・ガイド」の『**xlf_configure** ユーティリティーの直接実行』を参照してください。

ご使用条件およびライセンス情報が表示されます。

4. ご使用条件およびライセンス情報を読み、受諾します。ライセンス条件に同意する場合は、ご使用条件およびライセンス情報を受諾して、インストールを続けてください。

コンパイラ呼び出し用のシンボリック・リンクを `/usr/bin/` ディレクトリー内に作成することを確認するプロンプトが表示されます。

5. オプションでこのシンボリック・リンクを作成します。

注: このステップの代わりに、`PATH` 環境変数へのコンパイラ呼び出しを含むパスを追加できます。 33 ページの『コンパイラ呼び出しへのパスを組み込むように `PATH` 環境変数を設定する』を参照してください。

シンボリック・リンクの作成を選択すると、以下のリンクが `/usr/bin/` サブディレクトリー内に作成されます。

- `gxlC`
- `gxlC++`
- `gxlC`
- `xlC`
- `xlC++`
- `xlC`
- `xlC_r`
- `xlC++_r`
- `xlC_r`

注: 一部のコマンド・リンクは `/usr/bin/` に作成されません。これは、それらのリンクでユーザー定義呼び出しまたは `GCC` 呼び出しが削除されるため、またはコンパイラ呼び出しコマンドではないためです。以下のものが含まれます。

- `c89`、`c89_r`、`c99`、`c99_r`、`cc`、および `cc_r`
- `cleanpdf`、`mergepdf`、`new_install`、`resetpdf`、`showpdf`、`vac_configure`

すべてのパッケージが正常にインストールされると、以下の結果になります。

- インストールの成功を確認するためのメッセージが表示されます。
- 構成ファイルが生成されます。そのロケーションは `etc/opt/ibmcomp/vac/8.0/vac.cfg` です。以前に生成されたすべての構成ファイルは、名前変更されて、同じディレクトリーに保管されます。
- インストール・ログは、その永続的な場所 (`/opt/ibmcomp/vac/8.0/xlc_install.log`) に移動されます。

xlc_install オプション

`xlc_install`ユーティリティーでは、以下のオプションが提供されています。

-h インストール・ユーティリティー・ヘルプ・ページを表示します。

-rpmloc *rpmlocation_path*

すべてのコンパイラ・パッケージが置かれているパスを明示的に指定します。デフォルトの *rpmlocation_path* は `platform/rpms` で、これはインストール・ツールのパスに対する相対パスです。したがって、デフォルトの *rpmlocation_path* は以下のいずれかです。

- `./RHEL4/rpms` (RHEL4U2 にインストールする場合)

- ./SLES9/rpms (SLES9 SP2 にインストールする場合)
- ./SLES10/rpms (SLES10 にインストールする場合)
- ./Y-HPC/rpms (Y-HPC または YDL にインストールする場合)

注: ほとんどのユーザーの場合、インストール呼び出しで **-rpmloc** *rpmlocation_path* オプションは必要ありません。 CD または電子イメージから直接、ユーティリティを使用した場合、ユーティリティがパッケージのソース場所を自動的に判別します。

- U コンパイラーを、インストール・ユーティリティ・バージョンがサポートするバージョン.リリース.モディフィケーション-修正 (V.R.M-F) レベルに更新します。詳しくは、21 ページの『**xlc_install** ユーティリティを実行して、基本インストールを更新する』を参照してください。
- v デバッグ情報を表示します。
- vv より詳細なデバッグ情報を表示します。

第 3 章 上級者向けインストール

XL C/C++ for Linux は、デフォルトの位置に、11 ページの『第 2 章 基本インストール』で示された手順に従ってインストールするよう強くお勧めします。ただし次のようなカスタマイズされたシナリオでは、それ以外の手順を使用する必要があります。

- 同じシステムに複数のバージョンの XL C/C++ for Linux を維持したい場合。考えられるシナリオおよびインストール手順については、『XL C/C++ V8.0 をインストールして前のバージョンと共存させる』を参照してください。
- デフォルト・ロケーション以外の場所にインストール済みの XL C/C++ の既存のバージョンをアップグレードまたは更新したい場合。インストール手順については、17 ページの『XL C/C++ をデフォルト・ロケーション以外の場所にインストールする』を参照してください。
- 既にインストール済みのものをデフォルト・ロケーションから除去する前に、コンパイラーの新規更新を試してみたい場合。この場合、更新をデフォルト・ロケーション以外の場所にインストールする必要があります。手順については、17 ページの『XL C/C++ をデフォルト・ロケーション以外の場所にインストールする』を参照してください。

以上のようなシナリオではすべて、コンパイラーをインストールするのに **rpm** ユーティリティを使用する必要があります。**xl_install** ユーティリティは使用できません。コンパイラーをデフォルト・ロケーション以外の場所に正常にインストールし終えたなら、**new_install** または **vac_configure** ユーティリティを使用してコンパイラー環境を手動で構成する必要があります。手順については、25 ページの『第 5 章 XL C/C++ の構成 (上級者向け)』を参照してください。

XL C/C++ V8.0 をインストールして前のバージョンと共存させる

XL C/C++ V8.0 をインストールして、同じシステム上で前のバージョンの XL C/C++ と共存させるには、以下のオプションがあります。

- 複数のバージョンを同じ場所にインストールして、すべてのバージョンが最新のランタイム環境を使用するようにできます。XL C/C++ の異なるバージョンを長期間使用し続けたい場合は、このオプションをお勧めします。新しいバージョンが、古いバージョンのランタイム・パッケージを使用しない限り、複数のバージョンの XL C/C++ が同じ場所に共存できます。既存のバージョンがデフォルト・ロケーションにインストール済みの場合は、**xl_install** を使用して追加のバージョンをインストールしようとししないでください。古いファイルが除去されることになります。この場合、16 ページの『前のバージョンと同じロケーションに共存させる XL C/C++ V8.0 のインストール』の手順を使用してください。
- 各バージョンがそのバージョンに同梱出荷されたランタイム環境を使用するようにできます。より新しいバージョンに段階的にマイグレーションしたい場合は、このオプションをお勧めします。この場合、XL C/C++ V8.0 を XL C/C++ の他のすべてのバージョンと異なる場所にインストールする必要があります。そのため

には、17 ページの『XL C/C++ をデフォルト・ロケーション以外場所にインストールする』の手順の 1 つを実行してください。

前のバージョンと同じロケーションに共存させる XL C/C++ V8.0 のインストール

以下の手順は、XL C/C++ V8.0 のインストール方法、および同じロケーションにある既存のバージョンの XL C/C++ によって使用されるランタイム・パッケージの更新方法について説明しています。この手順は、次のことを前提としています。

- XL C/C++ V8.0 のインストール先は、すでに XL C/C++ V7.0 がインストールされているのと同じロケーションである (デフォルトでは、`/opt/ibmcomp/`)。
- 現行作業ディレクトリーには XL C/C++ V8.0 のすべてのパッケージが含まれるが、他の RPM パッケージは含まれない。
- 表 13 にリストされた既存のランタイム・コンポーネントのバージョンが、同じインストール・ロケーションにインストールされている (デフォルトでは、`/opt/ibmcomp/`)。

表 13. XL C/C++ SMP およびランタイム・パッケージ (RHEL4U2 または SLES9 SP2 のインストール用)

既存の XL C/C++ V7.0 のランタイム・パッケージ	新しい XL C/C++ V8.0 のランタイム・パッケージ
xlsmp.msg.rte-1.5.0-0	xlsmp.msg.rte-1.6.0-0
xlsmp.rte-1.5.0-0	xlsmp.rte-1.6.0-0
xlsmp.lib-1.5.0-0	xlsmp.lib-1.6.0-0
vacpp.rte-7.0.0-0	vacpp.rte-8.0.0-0

XL C/C++ V8.0 をインストールし、XL C/C++ V7.0 XL SMP およびランタイム・パッケージを更新するには、次の手順で行います。

1. 将来の依存エラーを回避するために、次のコマンドを発行して既存のランタイム・パッケージを除去します。

```
rpm -e vacpp.rte-7.0.0-0 --nodeps
rpm -e xlsmp.lib-1.5.0-0 --nodeps
rpm -e xlsmp.rte-1.5.0-0 --nodeps
rpm -e xlsmp.msg.rte-1.5.0-0 --nodeps
```

注: この例では、更新が適用されていないパッケージ名が示されています。更新が適用されている場合は、`-0` のフィックス・レベルが異なります。

2. 削除されたランタイム・パッケージを置き換えて、XL C/C++ V8.0 をインストールするために、次のコマンドを発行します。

```
rpm -ivh *.rpm
```

3. 念のため、既存のすべての構成ファイルのバックアップ・コピーを作成します。
4. 既存のすべての XL C/C++ V7.0 の構成ファイルを、次のように新しい構成パスを使用するように変更します。

```
# sed -e "s/xlsmp%/1%.5/xlsmp%/1%.6/g" < /etc/installation_path/vac/7.0/vac.cfg >
/etc/installation_path/vac/7.0/vac.cfg.new
# mv /etc/installation_path/vac/7.0/vac.cfg.new
/etc/installation_path/vac/7.0/vac.cfg
```

ここで、*installation_path* は、すべての XL C/C++ パッケージがインストールされているロケーションを示します (デフォルトでは、*/opt/ibmcmp/*)。

注: XL C/C++ V7.0 のパスは変更されません。

5. 次のように、デフォルトの XL C/C++ V8.0 構成ファイルを生成します。

```
# /installation_path/vac/8.0/bin/vac_configure -gcc /usr -gcc64 /usr
-ibmcmp /installation_path/ /opt/ibmcmp/vac/8.0/etc/vac.base.cfg
-o /etc/installation_path/vac/8.0/vac.cfg
```

ここで、*installation_path* は、すべての XL C/C++ パッケージがインストールされているロケーションを示します (デフォルトでは、*/opt/ibmcmp/*)。

XL C/C++ をデフォルト・ロケーション以外の場所にインストールする

この節では、考えられる 2 つのインストール・シナリオでの手順を示します。

- すべてのコンパイラー・パッケージを単一のデフォルト・ロケーション以外の場所にインストールできます。例えば、すべてのパッケージをデフォルト・ディレクトリー */opt/ibmcmp/* にインストールするのではなく、*/home/mydirectory/* などの選択したディレクトリーにインストールできます。このための手順は、『すべてのパッケージを単一のデフォルト・ロケーション以外の場所にインストールする』に示しています。
- 特殊な状態の場合のみ、1 グループのコンパイラー・パッケージを複数の異なる場所にインストールできます。例えば、すべてのコンパイラー・ライブラリー・パッケージを 1 つのディレクトリーにインストールし、ランタイム環境パッケージを別のディレクトリーにインストールしたりできます。ただし、一定のパッケージをまとめて同じライブラリーにインストールしなければならない場合があります。パッケージを複数のデフォルトではないディレクトリーにインストールする場合の規則および手順は、18 ページの『パッケージを複数のデフォルト・ロケーション以外の場所にインストールする』に示しています。

いずれのシナリオの場合にも、コンパイラーの前のバージョンをシステムからアンインストールしてから、新しいバージョンをインストールするようにお勧めします。アンインストール手順については、35 ページの『第 7 章 XL C/C++ のアンインストール』を参照してください。

すべてのパッケージを単一のデフォルト・ロケーション以外の場所にインストールする

すべてのコンパイラー・パッケージを単一のデフォルトではないディレクトリーにインストールするには、現行作業ディレクトリーに XL C/C++ V8.0 for Linux 用のすべてのパッケージが含まれ、他の RPM パッケージが含まれないようにします。現行作業ディレクトリーから以下のコマンドを使用します。

```
rpm -ivh *.rpm --prefix installation_path
```

ここで、*installation_path* は、*/opt/ibmcmp/* 以外のディレクトリーです。

パッケージを複数のデフォルト・ロケーション以外の場所にインストールする

特殊な状態の場合のみ、別のパッケージを別の場所にインストールすることをお勧めします。

注: パッケージを異なるサブディレクトリーにインストールしたい場合、どのパッケージも `/opt/ibmcmp/` ディレクトリーにはインストールしないでください。

表 14 には、同一のディレクトリーにインストールしなければならないパッケージ、および、任意のディレクトリーにインストールできるパッケージについての情報が提供されています。

表 14. パッケージを複数のデフォルト・ロケーション以外の場所にインストールする場合の規則

パッケージ名	パッケージの説明	デフォルト・ロケーション以外の場所へのインストールの規則
xlsmp.msg.rte	XL SMP ランタイム・メッセージ	XL SMP パッケージは、すべて同じ場所にインストールされなければならない。本書の残りの部分では、この場所を指すのに名前 <code>xlsmp_rte_path</code> が使用されています。
xlsmp.rte	XL SMP ランタイム動的ライブラリー	
xlsmp.lib	XL SMP ランタイム静的ライブラリー	
xlmass.lib	IBM Mathematical Acceleration Subsystem (MASS) ライブラリー	任意のロケーション。本書の残りの部分では、この場所を指すのに名前 <code>xlmass_path</code> が使用されています。
vacpp.rte	XL C/C++ ランタイム環境	XL C/C++ ランタイム・パッケージは、すべて同じ場所にインストールされなければならない。本書の残りの部分では、この場所を指すのに名前 <code>xl_rte_path</code> が使用されています。
vacpp.rte.lnk	XL C/C++ ランタイム環境リンク	
vac.lic	XL C/C++ ライセンス	任意のロケーション。本書の残りの部分では、この場所を指すのに名前 <code>lic_path</code> が使用されています。
vac.lib	XL C コンパイラー・ライブラリー	XL C/C++ コンパイラーおよびライブラリー・パッケージは、すべて同じ場所にインストールされなければならない。本書の残りの部分では、この場所を指すのに名前 <code>xlcmp_path</code> が使用されています。
vac.cmp	XL C コンパイラー・ファイル	
vacpp.lib	XL C++ コンパイラー・ライブラリー	上記の他のコンパイラー・パッケージおよびライブラリー・パッケージと同じ場所にインストールされなければならない。
vacpp.cmp	XL C++ コンパイラー・ファイル	
vacpp.help	XL C/C++ の資料	任意の場所 (オプション) 本書の残りの部分では、この場所を指すのに名前 <code>doc_path</code> が使用されています。

表 14. パッケージを複数のデフォルト・ロケーション以外の場所にインストールする場合の規則 (続き)

パッケージ名	パッケージの説明	デフォルト・ロケーション以外の場所へのインストールの規則
vacpp.samples	XL C/C++ サンプル	任意の場所 (オプション) 本書の残りの部分では、この場所を指すのに名前 <i>smpls_path</i> が使用されています。

再配置可能 RPM パッケージをデフォルト・ロケーション以外の任意の場所にインストールするには、デフォルト以外のディレクトリーにインストールしたいパッケージのグループごとに以下のコマンドを発行します。

```
rpm -ivh package --prefix package_installation_path
```

ここで、*package_installation_path* は、*/opt/ibmcmp/* 以外のディレクトリーで、18 ページの表 14 にリストされた適切なパスの 1 つに対応しています。

例 (SLES10、Y-HPC、YDL): XL C/C++ V8.0 for Linux を複数のデフォルトではないディレクトリーにインストールする

XL C/C++ V8.0 for Linux のインストール時の依存関係のエラーを回避するために、以下のコマンドを指定された順序で発行してください。

```
rpm -ivh xlsmp.msg.rte-1.6.1-0.ppc64.rpm --prefix xlsmp_rt_path
rpm -ivh xlsmp.rte-1.6.1-0.ppc64.rpm --prefix xlsmp_rt_path
rpm -ivh xlsmp.lib-1.6.1-0.ppc64.rpm --prefix xlsmp_rt_path
rpm -ivh xlmass.lib-4.3.1-0.ppc64.rpm --prefix xlmass_path

rpm -ivh vacpp.rte-8.0.1-0.ppc64.rpm --prefix xlrte_path
rpm -ivh vacpp.rte.lnk-8.0.1-0.ppc64.rpm --prefix xlrte_path
rpm -ivh vac.lic-8.0.1-0.ppc64.rpm --prefix lic_path
rpm -ivh vac.lib-8.0.1-0.ppc64.rpm --prefix xlcmp_path
rpm -ivh vac.cmp-8.0.1-0.ppc64.rpm --prefix xlcmp_path
rpm -ivh vacpp.lib-8.0.1-0.ppc64.rpm --prefix xlcmp_path
rpm -ivh vacpp.cmp-8.0.1-0.ppc64.rpm --prefix xlcmp_path
```

サンプル・プログラムおよび製品資料パッケージは、他の RPM パッケージとの依存関係がないため、以下のコマンドを使用して任意の順序でインストールできます。

```
rpm -ivh vacpp.help-8.0.1-0.ppc64.rpm --prefix doc_path
rpm -ivh vacpp.samples-8.0.1-0.ppc64.rpm --prefix smpls_path
```

第 4 章 更新のインストール

XL C/C++ V8.0 for Linux の更新によって、製品の 1 つまたは複数のフィックスが提供されます。更新は、以下のサポート Web サイトからダウンロードできます。

<http://www.ibm.com/software/awdtools/xlcpp/support>

更新パッケージは、すべて tar.gz (または圧縮) フォーマットで提供され、付随する更新のみをインストールするようにカスタマイズされた **xlc_install** ユーティリティーのバージョンを含みます。システムにインストールされている XL C/C++ V8.0 for Linux のいずれかのバージョン (以前の更新を含む) があれば、最新の更新を適用できます。デフォルト・ロケーションのインストールに更新を適用する場合は、『**xlc_install** ユーティリティーを実行して、基本インストールを更新する』の手順に従ってください。デフォルト・ロケーション以外の場所のインストール済み環境に更新を適用する場合は、17 ページの『XL C/C++ をデフォルト・ロケーション以外の場所にインストールする』のいずれかの手順を使用する必要があります。

既存のバージョンをシステムから除去する前に、コンパイラーの新規の更新を試してみたい場合は、新規の更新をデフォルト・ロケーション以外の場所にインストールする必要があります。古いバージョンを新しい更新で置き換えたいことを確認した後で、その更新パッケージと共に提供される **xlc_install** ユーティリティー実行すると、以下のすべてのことが行われます。

1. 新規の更新がデフォルト・ロケーションではない場所から除去されます。
2. 古いバージョンがデフォルト・ロケーションから除去されます。

注: **vac.lic** パッケージは、次のステップの実行時に必要であるため、アンインストールしないようにしてください。

3. 新規の更新がデフォルト・ロケーションにインストールし直されます。

xlc_install ユーティリティーを使用して新規の更新をインストールする方法の説明は、『**xlc_install** ユーティリティーを実行して、基本インストールを更新する』を参照してください。

xlc_install ユーティリティーを実行して、基本インストールを更新する

以下の条件がすべて満たされると、**xlc_install** ユーティリティーを使用して XL C/C++ V8.0 for Linux を更新できます。

- XL C/C++ V8.0 for Linux の基本バージョンが既に /opt/ibmcomp/ ディレクトリー内に正常にインストールされている。
- 更新パッケージ (tar.gz 形式) が解凍され、システムの /home/root/ ディレクトリー内にアンパックされている。

xlc_install ユーティリティーを実行して更新を適用する場合は、以下のことが行われます。

- 前提のソフトウェア・パッケージがすべてチェックされます。
- XL C/C++ V8.0 for Linux パッケージがアンインストールされます。

- 更新されたコンパイラー・パッケージがデフォルト・ロケーションにインストールされます。
- **new_install** ユーティリティーが自動的に呼び出されます。このユーティリティーにより、ライセンス・ファイルがインストールされ、古い構成ファイルが名前変更されて、新しい構成ファイルが生成されます。
- オプションでコンパイラー呼び出しコマンドへのシンボリック・リンクが `/usr/bin/` に作成されます。
- インストール・ログが `/tmp/` ディレクトリーに生成されます。

xlc_install ユーティリティーを実行して IBM XL C/C++ Advanced Edition V8.0 for Linux の更新を適用する方法

1. 次を入力して、更新パッケージを解凍したディレクトリーに変更します。

```
# cd /home/root/update/xlc/mmmYYYY
```

ここで、*mmmYYYY* は更新出荷日付の月と年です。(例えば、*jun2006* は、2006 の 6 月の出荷日を示します。)

2. 次のコマンドを発行します。

```
# ./xlc_install -U
```

xlc_install に指定できる追加の引数については、13 ページの『**xlc_install** オプション』を参照してください。

前にインストール済みの XL C/C++ V8.0 for Linux パッケージをアンインストールすることを確認するプロンプトが表示されます。

3. 期限切れのパッケージのアンインストールを進めることを確認します。

XL C/C++ V8.0 for Linux とともに前にインストール済みの XL SMP および IBM MASS パッケージをアンインストールすることを確認するプロンプトが表示されます。

4. 既存の XL SMP および IBM MASS パッケージのアンインストールを進めることを確認します。

注: **xlc_install** は、これらのパッケージを既存の場所からアンインストールして、デフォルト・ロケーション (`/opt/ibmcmp/`) にインストールし直します。したがって、パッケージが前にデフォルト・ロケーション以外の場所に XL Fortran for Linux のインストールの一部としてインストール済みである場合は、**xlf_configure** を実行して、XL Fortran コンパイラーを再構成し、これらのパッケージのデフォルト・ロケーションを示す必要があります。手順については、「*XL Fortran Advanced Edition V10.1 for Linux* インストール・ガイド」の『**xlf_configure** ユーティリティーの直接実行』を参照してください。

ご使用条件およびライセンス情報が表示されます。

5. ライセンス情報およびご使用条件を受諾します。

コンパイラー呼び出し用のシンボリック・リンクを `/usr/bin/` ディレクトリー内に作成することを確認するプロンプトが表示されます。

6. オプションでこのシンボリック・リンクを作成します。

注: このステップの代わりに、PATH 環境変数へのコンパイラ呼び出しを含むパスを追加できます。 33 ページの『コンパイラ呼び出しへのパスを組み込むように PATH 環境変数を設定する』を参照してください。

シンボリック・リンクの作成を選択すると、以下のリンクが /usr/bin/ サブディレクトリー内に作成されます。

- gxc
- gxc++
- gxC
- xc
- xc++
- xC
- xc_r
- xc++_r
- xC_r

注: 一部のコマンド・リンクは /usr/bin/ に作成されません。これは、それらのリンクでユーザー定義呼び出しまたは GCC 呼び出しが削除されるため、またはコンパイラ呼び出しコマンドではないためです。以下のものが含まれます。

- c89、c89_r、c99、c99_r、cc、および cc_r
- cleanpdf、mergepdf、new_install、resetpdf、showpdf、vac_configure

7. 前に生成された構成ファイルをカスタマイズした場合は、手動で /etc/opt/ibmcomp/vac/8.0/vac.cfg を編集し、新規に生成された構成ファイル内にそれらの変更内容を複製してください。

第 5 章 XL C/C++ の構成 (上級者向け)

次のいずれかの条件が該当する場合、XL C/C++ for Linux の実行前に、コンパイラーを構成 (または再構成) する必要があります。

- **xlc_install** を使用して、コンパイラーをインストールしていない。
- コンパイラーがデフォルト・ロケーション以外の場所にインストールされているか、インストール後にコンパイラー・コンポーネントが再配置された。

コンパイラーによって提供される構成ツールには、**new_install** と **vac_configure** の 2 つがあり、いずれもインストール後は *installation_path/vacpp/8.0/bin/*ディレクトリーに入れられます。

以下の条件がすべて満たされた場合は、**new_install** ユーティリティーを使用してコンパイラーを構成するようにお勧めします。

- XL C/C++ for Linux の 1 つのバージョンのみがご使用のシステムにインストールされている。
- GCC の 1 つのバージョンのみがご使用のシステムにインストールされていて、PATH 環境変数内で検索できる。
- root または管理者特権を持っている。
- 構成ファイルをデフォルト・ディレクトリー */etc/opt/ibmcomp/vac/8.0/* に生成したい。

詳細な説明については、26 ページの『**new_install** ユーティリティーを実行する』を参照してください。

以下のいずれかの条件が該当する場合にのみ、**vac_configure** ユーティリティーを直接呼び出すようにしてください。

- XL C/C++ for Linux の複数のバージョンがシステムにインストールされている。
- **new_install** コマンドからエラーが受け取られた。(37 ページの『第 8 章 インストールおよび構成のトラブルシューティング』を参照してください。)
- 生成された構成ファイルをデフォルト・ロケーション以外の場所に置きたい。
- GCC の複数のバージョンがシステム上にインストールされており、どの GCC バージョンを構成ファイル内で参照したいかを指定する必要がある。

注: **vac_configure** を使用してコンパイラーを構成する場合、書き込み許可を持っている場所に出力構成ファイル *vac.cfg* を書き込むことができます。この場合は、root または管理者特権が必要ありません。

詳細な説明については、26 ページの『**vac_configure** ユーティリティーを直接実行する』を参照してください。

new_install ユーティリティを実行する

new_install ユーティリティは、以下のことを行います。

- 既存の構成ファイルをすべてバックアップします。
- コンパイラー・パッケージへのパスおよび PATH 環境変数内の 32 ビット GCC (*gcc32path*) と 64 ビット GCC (*gcc64path*) へのパスについて RPM データベースを照会し、取得した値を使用して **vac_configure** ユーティリティを実行します。
- ライセンス・ファイルをインストールします。
- 構成ファイルをデフォルト・ロケーション (*/etc/opt/ibmcmp/vac/8.0/vac.cfg*) に生成します。

new_install ユーティリティの実行方法

1. コンパイラーの実行可能ファイルを含むディレクトリに変更します。

```
cd installation_path/vacpp/8.0/bin/
```

ここで、*installation_path* は、コンパイラー・パッケージのインストール・ロケーションです。コンパイラーがデフォルト・ロケーションにインストールされている場合、*installation_path* は */opt/ibmcmp/* です。

2. 次のコマンドを実行します。

```
./new_install
```

3. ご使用条件およびライセンス情報を読み、受諾します。ライセンス条件に同意する場合は、ご使用条件およびライセンス情報を受諾します。

vac_configure ユーティリティを直接実行する

コンパイラーが正常にインストールされていれば、**vac_configure** ユーティリティを使用して、構成ファイルを生成することができます。

vac_configure ユーティリティの実行方法

1. コンパイラーの実行可能ファイルを含むディレクトリに変更します。

```
cd installation_path/vacpp/8.0/bin/
```

ここで、*installation_path* は、コンパイラー・パッケージのインストール・ロケーションです。コンパイラーがデフォルト・ロケーションにインストールされている場合、*installation_path* は */opt/ibmcmp/* です。

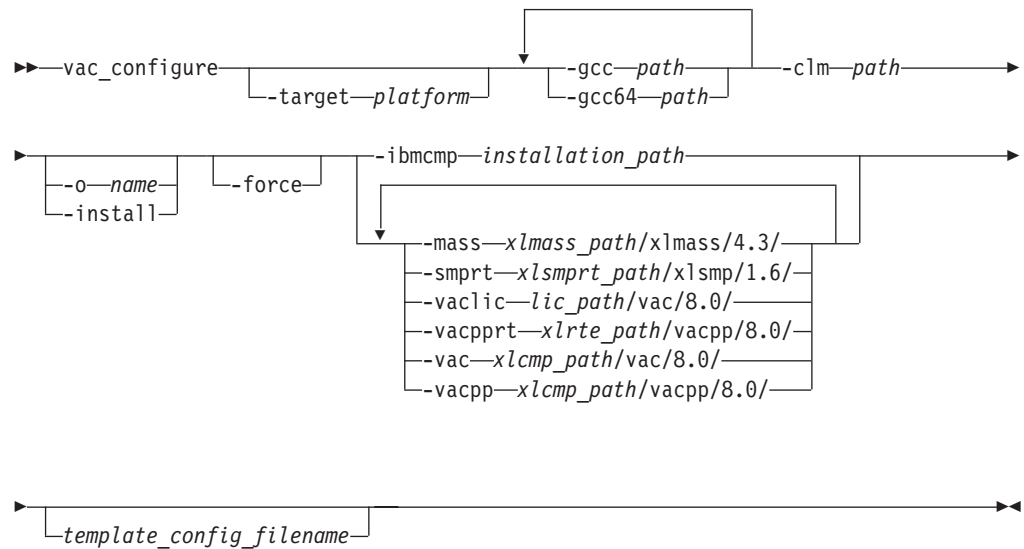
2. 次のコマンドを実行します。

```
./vac_configure options
```

vac_configure コマンドに必要な引数については、次の節を参照してください。

vac_configure オプション

vac_configure コマンドの構文は、以下のとおりです。



ここで、

-target platform

オペレーティング・システム・プラットフォームを指定します。有効な値は以下のとおりです。

- sles
- rhel
- yhpc

値を指定しない場合は、デフォルトでホスト・オペレーティング・システムと対応するプラットフォームとなります。

-gcc path

固有 GCC bin/ ディレクトリーがインストールされているパスを指定します。例えば、GCC コマンドが `/usr/bin/gcc` の場合、次のように指定します。

```
-gcc /usr
```

-gcc64 path

固有 64 ビット GCC bin/ ディレクトリーがインストールされているパスを指定します。例えば、64 ビット GCC コマンドが `/usr/bin/gcc -m64` の場合、次のように指定します。

```
-gcc64 /usr
```

-clm path

共通ライセンス・マネージャーの `license.dat` ファイルが存在しているパスを指定します。デフォルトでは、これは `/opt/clm_ibm` です。

-o file_name

生成する構成ファイルの名前とパスを指定します。デフォルトでは、出力はディスプレイのみに書き込まれます。

-install 構成ファイルを `/etc/opt/ibmcmp/vac/8.0/vac.cfg` として生成します。デフォルトでは、出力はディスプレイのみに書き込まれます。

-force vac_configure ユーティリティーに既存の出力ファイルを **-o** または **-install** オプションで指定された名前とパスで上書きさせます。デフォルトでは、**force** を使用しないと、指定されたファイルがすでに存在する場合は、**vac_configure** がエラー・メッセージを出して停止します。

-ibmcmp installation_path

XL C/C++ パッケージのすべてがインストールされているパスを指定します (すべてのパッケージが同じパスにインストールされている場合)。デフォルトでは、このパスは `/opt/ibmcmp/` です。

-mass xlmass_path/xlmass/4.3/

`xlmass.lib` パッケージがインストールされているパスを指定します。デフォルトでは、絶対パスは `/opt/ibmcmp/xlmass/4.3` です。

-smpprt xlsmpprt_path/xlsmp/1.6/

`xlsmp.msg.rte`、`xlsmp.rte`、および `xlsmp.lib` パッケージがインストールされているパスを指定します。デフォルトでは、絶対パスは `/opt/ibmcmp/xlsmp/1.6` です。

-vaclic lic_path/vac/8.0/

`vac.lic` パッケージがインストールされているパスを指定します。デフォルトでは、絶対パスは `/opt/ibmcmp/vac/8.0/` です。

-vacpprt xlrte_path/vacpp/8.0/

`vacpp.rte` および `vacpp.rte.lnk` パッケージがインストールされているパスを指定します。デフォルトでは、絶対パスは `/opt/ibmcmp/vacpp/8.0/` です。

-vac xlcmp_path/vac/8.0/

`vac.cmp` および `vac.lib` パッケージがインストールされているパスを指定します。デフォルトでは、絶対パスは `/opt/ibmcmp/vac/8.0/` です。

-vacpp xlcmp_path/vacpp/8.0/

`vacpp.cmp` および `vacpp.lib` パッケージがインストールされているパスを指定します。デフォルトでは、絶対パスは `/opt/ibmcmp/vacpp/8.0/` です。

template_config_file_name

構成ファイルを構成するために使用される入力ファイル。デフォルトでは、これは `/opt/ibmcmp/vac/8.0/etc/vac.base.cfg` です。`vac.cmp` パッケージを `xlcmp_path` に再配置したが、デフォルト・テンプレートを使用したい場合は、次のように指定します。

`xlcmp_path/vac/8.0/etc/vac.base.cfg`.

第 6 章 XL C/C++ のインストール後の処置

コンパイラーのインストール後、検査およびセットアップ手順を実行する必要があります (あるいは必要になる場合があります)。これらについては、以下の各節で説明しています。

- 『インストール済みパッケージの照会』：この節は、すべてのユーザーが対象となります。
- 30 ページの『インストールのテスト』：この節は、すべてのユーザーが対象となります。
- 31 ページの『マニュアル・ページの使用可能化』：この節は、すべてのユーザーが対象となります。
- 31 ページの『エラー・メッセージの使用可能化』：この節は、en_US 以外のロケールまたは言語エンコードが使用されているシステムのユーザーのみが対象となります。
- 32 ページの『呼び出しコマンド用の環境のセットアップ』：この節は、製品のインストールまたは更新に **xl_c_install** を使用しなかったユーザー、または **xl_c_install** によるインストール・プロセスでシンボリック・リンクを作成しなかったユーザーのみが対象となります。

インストール済みパッケージの照会

個々のパッケージについて照会するには、以下のようなコマンドを発行してください。

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

結果は次のようになるはずです。

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

ここで、*V.R.M-F* は、システム上にインストールされたコンパイラーのバージョン、リリース、モディフィケーション-修正レベルを表します。

インストールが成功しなかった場合は、そのパッケージがインストールされなかったことを示すメッセージを受け取ります。

すべてのコンパイラー・パッケージのインストールを確認するには、以下のコマンドを発行してください。

```
rpm -qa | grep -e vac -e xlsmp -e xlmass
```

その結果は、2 ページの表 4 にリストされているパッケージのすべてを含むリストであるはずです。表にリストされているパッケージのいずれも適切にインストールされなかった場合、コマンドからの出力はありません。

インストールのテスト

製品インストールおよび重要な検索パスをテストするには、サンプル・アプリケーションを作成して実行します。

基本例: "Hello World" の作成および実行

1. 以下の C プログラムを作成し、ソース・ファイルの名前を `hello.c` にします。

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

2. プログラムをコンパイルします。

- 短形式の呼び出しコマンドがセットアップされている場合は、以下のコマンドを入力します。

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

- 短形式の呼び出しコマンドがセットアップされていない場合は、以下のコマンドを入力します。

```
/opt/ibmcomp/vacpp/8.0/bin/xlc hello.c -o hello
```

3. 次のコマンドを入力して、プログラムを実行します。

```
./hello
```

その結果は、"Hello World!" のはずです。

4. 次のコマンドを入力して、プログラムの終了コードを確認します。

```
echo $?
```

結果は 0 になるはずです。

5. 以下の C++ プログラムを作成し、ソース・ファイルの名前を `hello.cpp` にします。

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

6. プログラムをコンパイルします。

- 短形式の呼び出しコマンドがセットアップされている場合は、以下のコマンドを入力します。

```
xlc++ hello.cpp -o hello
```

- 短形式の呼び出しコマンドがセットアップされていない場合は、以下のコマンドを入力します。

```
/opt/ibmcomp/vacpp/8.0/bin/xlc++ hello.cpp -o hello
```

7. 次のプログラムを実行します。

```
./hello
```

その結果は、"Hello World!" のはずです。

8. プログラムの終了コードを調べます。

```
echo $?
```

結果は "0" になるはずです。

マニュアル・ページの使用可能化

マニュアル・ページは、コンパイラー呼び出しコマンドおよびその他のユーティリティー (コンパイラーと同梱出荷されるもの) に対して提供されています。

XL C/C++ V8.0 マニュアル・ページは、以下の言語ロケールをサポートします。

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

ただし、コンパイラー提供のマニュアル・ページを読むには、その絶対ディレクトリー・パスを `MANPATH` 環境変数に追加する必要があります。コマンドは、使用中の Linux シェルによって決まります。

Bourne、Korn、または BASH シェルを使用して `MANPATH` 環境変数を設定するには、以下のコマンドを使用します。

```
export MANPATH=installation_path/vacpp/8.0/man:$MANPATH
```

C シェルを使用して `MANPATH` 環境変数を設定するには、以下のコマンドを使用します。

```
setenv MANPATH installation_path/vacpp/8.0/man:$MANPATH
```

ここで、*installation_path* は、XL C/C++ パッケージをインストールした場所 (デフォルトでは、`/opt/ibmcmp/`) です。

注: この変数をすべてのユーザーに適用されるように Bourne、Korn、または BASH シェル内で設定するには、コマンドをファイル `/etc/profile` に追加します。特定のユーザーにのみ設定するには、コマンドをそのユーザーのホーム・ディレクトリーのファイル `.profile` に追加します。C シェルでは、コマンドをファイル `/etc/csh.cshrc` に追加します。特定のユーザーにのみ設定するには、コマンドをそのユーザーのホーム・ディレクトリーのファイル `.cshrc` に追加します。ユーザーがログインするたびに、環境変数が設定されます。

エラー・メッセージの使用可能化

システムで `en_US` のロケールとエンコードが使用されている場合、コンパイラー・メッセージ・カタログが正しく表示されるように自動的に構成されます。インストールと構成方法が基本的であるか、上級者向けであるかに関係ありません。ただし、システムがサポートされる他のロケール (サポートされる言語ロケールのリストについては、2 ページの『各国語サポート』を参照) を使用する場合は、インストール後にコンパイラーおよびランタイム関数が適切なメッセージ・カタログを検索できるように、`NLSPATH` 環境変数を設定する必要があります。システムが

en_US ロケールを使用するが、ランタイムがデフォルト・ロケーション以外の場合にインストールされている場合は、NLSPATH 環境変数を設定する必要があります。

NLSPATH 環境変数を設定するコマンドは、使用中のシェルに依存します。

Bourne、Korn、または BASH シェルを使用中の場合は、以下のコマンドを使用します。

```
export NLSPATH=$NLSPATH:  
    xlsmprrt_path/msg/%L/%N:  
    xlrte_path/msg/%L/%N:  
    xlcmp_path/vacpp/8.0/msg/%L/%N
```

C シェルを使用中の場合は、以下のコマンドを使用してください。

```
setenv NLSPATH $NLSPATH:  
    xlsmprrt_path/msg/%L/%N:  
    xlrte_path/msg/%L/%N:  
    xlcmp_path/vacpp/8.0/msg/%L/%N
```

ここで、

- *xlsmprrt_path* は、XL SMP パッケージのインストール・ロケーションです。デフォルトでは、*/opt/ibmcmp/* です。
- *xlrte_path* は、XL C/C++ ランタイム・パッケージのインストール・ロケーションです。デフォルトでは、*/opt/ibmcmp/* です。
- *xlcmp_path* は、XL C/C++ コンパイラー・パッケージのインストール・ロケーションです。デフォルトでは、*/opt/ibmcmp/* です。

注: この変数をすべてのユーザーに適用されるように Bourne、Korn、または BASH シェル内で設定するには、コマンドをファイル */etc/profile* に追加します。特定のユーザーにのみ設定するには、コマンドをそのユーザーのホーム・ディレクトリーのファイル *.profile* に追加します。C シェルでは、コマンドをファイル */etc/csh.cshrc* に追加します。特定のユーザーにのみ設定するには、コマンドをそのユーザーのホーム・ディレクトリーのファイル *.cshrc* に追加します。ユーザーがログインするたびに、環境変数が設定されます。

呼び出しコマンド用の環境のセットアップ

xlc_install ユーティリティを使用してコンパイラーをインストールし、その時にシンボリック・リンクの作成を選択した場合、呼び出しコマンド用の環境はすでにセットアップされています。この節の手順を実行しないでください。

コンパイラーをインストールしたときにシンボリック・リンクの作成を選択せず、絶対パスを指定せずにコンパイラーを呼び出せるようにしたい場合は、以下の作業のいずれかを実行する必要があります。

- 33 ページの『コンパイラー呼び出しへのパスを組み込むように PATH 環境変数を設定する』に示されているように、PATH 環境変数を設定する。
- 33 ページの『コンパイラー呼び出しへのシンボリック・リンクの作成』に示されているように、コンパイラー呼び出しコマンドへのシンボリック・リンクを作成する。

コンパイラ呼び出しへのパスを組み込むように PATH 環境変数を設定する

完全パスを入力せずに XL C/C++ for Linux のコマンドを使用するために、コンパイラ呼び出しの場所を PATH 環境変数に追加できます。

Bourne、Korn、または BASH シェルを使用中の場合は、以下のコマンドを使用します。

```
export PATH=$PATH:installation_path/vacpp/8.0/bin/
```

C シェルを使用中の場合は、以下のコマンドを使用してください。

```
setenv PATH $PATH:installation_path/vacpp/8.0/bin/
```

ここで、*installation_path* は、コンパイラ・パッケージをインストールした場所 (デフォルトでは、*/opt/ibmcomp/*) です。

注: この変数をすべてのユーザーに適用されるように Bourne、Korn、または BASH シェル内で設定するには、コマンドをファイル */etc/profile* に追加します。特定のユーザーにのみ設定するには、コマンドをそのユーザーのホーム・ディレクトリーのファイル *.profile* に追加します。C シェルでは、コマンドをファイル */etc/csh.cshrc* に追加します。特定のユーザーにのみ設定するには、コマンドをそのユーザーのホーム・ディレクトリーのファイル *.cshrc* に追加します。ユーザーがログインするたびに、環境変数が設定されます。

コンパイラ呼び出しへのシンボリック・リンクの作成

完全パスを入力せずにコンパイラを使用するために、

installation_path/vacpp/8.0/bin/ ディレクトリーに含まれる特定の呼び出し用に、シンボリック・リンクを */usr/bin/* ディレクトリーに作成できます。

xlc_install を実行したときに、上記の操作を行なわなかった場合は、以下のコンパイラ呼び出し用のシンボリック・リンクを作成できます。

- gxc
- gxlc++
- gxlc
- xlc
- xlc++
- xlc
- xlc_r
- xlc++_r
- xlc_r

一部の呼び出しへのリンクはお勧めできません。これは、それらのリンクでユーザー定義呼び出しまたは GCC 呼び出しが削除されるため、またはコンパイラ呼び出しコマンドではないためです。これらは以下のコマンドを含みます。

- c89、c89_r、c99、c99_r、cc、および cc_r
- cleanpdf、mergepdf、new_install、resetpdf、showpdf、vac_configure

以下のコマンドを使用して、シンボリック・リンクを作成します。

```
ln -s installation_path/vacpp/8.0/bin/invocation /usr/bin/invocation
```

ここで、

- *installation_path* は、コンパイラー・パッケージをインストールした場所 (デフォルトでは、/opt/ibmcomp/) です。
- *invocation* は、*installation_path*/vacpp/8.0/bin/ 内のコンパイラー呼び出しの 1 つです (例えば、**xlc++**)。

基本例: xlc コンパイラー呼び出しに対するシンボリック・リンクの作成

この例では、XL C/C++ V8.0 for Linux 全体がデフォルト・ロケーション /opt/ibmcomp/ にインストール済みであることを前提としています。

```
ln -s /opt/ibmcomp/vacpp/8.0/bin/xlc /usr/bin/xlc
```

第 7 章 XL C/C++ のアンインストール

XL C/C++ は、スタンドアロンのアンインストール・ツールを提供しません。XL C/C++ をアンインストールするには、Linux **rpm** ユーティリティを使用する必要があります。

注:

1. コンパイラーをアンインストールするには、root ユーザー・アクセスが必要です。
2. パッケージをアンインストールするときは常に、そのパッケージの *V.R.M-F* (バージョン.リリース.モディフィケーション-修正レベル) を指定してください。
3. 必ず、パッケージがインストールされた順序の逆順でアンインストールしてください。つまり、最後にインストールしたパッケージを、最初に除去することになります。例外: サンプル・プログラムおよび製品資料のパッケージ間には、依存関係がありません。これらのパッケージは、任意の順序で除去することができます。
4. 他のパッケージが必要とするパッケージは、アンインストールできません。例えば、`xlsmp.rte` は、同じシステム上に XL Fortran Advanced Edition V10.1 for Linux もインストールされている場合は、共用コンポーネントになります。
5. `new_install` または `vac_configure` によって生成された構成ファイルは、アンインストール・コマンドで除去されません。

例 (SLES10、Y-HPC、YDL): XL C/C++ V8.0 for Linux のアンインストール

この例の内容は、以下のとおりです。

- コンパイラー・パッケージは、*V.R.M-F* が 8.0.1-0 です。
- IBM MASS ライブラリー・パッケージは、*V.R.M-F* が 4.3.1-0 です。
- XL SMP ライブラリー・パッケージは、*V.R.M-F* が 1.6.1-0 です。

XL C/C++ V8.0 for Linux をアンインストールするには、以下のコマンドを同じ順序で発行してください。

```
rpm -e vacpp.cmp-8.0.1-0
rpm -e vacpp.lib-8.0.1-0
rpm -e vac.cmp-8.0.1-0
rpm -e vac.lib-8.0.1-0
rpm -e vac.lic-8.0.1-0
rpm -e vacpp.rte.lnk-8.0.1-0
rpm -e vacpp.rte-8.0.1-0

rpm -e xlmass.lib-4.3.1-0
rpm -e xlsmp.lib-1.6.1-0
rpm -e xlsmp.rte-1.6.1-0
rpm -e xlsmp.msg.rte-1.6.1-0
```

以下のコマンドは、任意の順序で発行することができます。

```
rpm -e vacpp.samples-8.0.1-0  
rpm -e vacpp.help-8.0.1-0
```

第 8 章 インストールおよび構成のトラブルシューティング

インストール・ユーティリティーは、インストール・プロセスの初期段階で新規ログ・ファイルを `/tmp/` 内に作成します。一時ログ・ファイルは、一意的に名前が付けられます。

インストールが正常に完了した後で、ログ・ファイルはデフォルト・インストール・ロケーションに移動され、それ以降参照できるようになります。インストールが失敗した場合は、インストール・ログはそのまま `/tmp/` ディレクトリー内に残ります。インストールの成功または失敗に関係なく、対応するインストール・ログのファイル名が標準出力の一部として表示されます。

この節の情報を使用すると、XL C/C++ のインストールおよび構成時に起こる可能性がある問題に対応する際に役立ちます。

エラー・メッセージおよび推奨処置

コンパイラーは、ユーザーがエラー条件を認識して対応する際に役立つメッセージを生成します。本節には、推奨される応答が準備されています。

指定されたディレクトリー `rpmlocation_path` が存在しない

シナリオ

xlc_install ユーティリティーを実行して、コンパイラーをデフォルト・ロケーションにインストールする際に、以下のエラー・メッセージを受け取ります。

```
ERROR: The specified directory, "rpmlocation_path", does not exist.
```

アクション

既存のコンパイラー・パッケージの場所を正しく指定したかを確認してください。 **xlc_install** ユーティリティーをインストール・イメージで指定された以外の場所に移動した場合、**-rpmloc rpmlocation_path** オプションを使用する必要があります。詳しくは、13 ページの『**xlc_install** オプション』を参照してください。

`rpmlocation_path` に . . . が含まれていない

シナリオ

xlc_install ユーティリティーを実行して、コンパイラーをデフォルト・ロケーションにインストールする際に、以下のエラー・メッセージを受け取ります。

```
ERROR: rpmlocation_path does not contain all of the RPM packages  
for the XL compiler.
```

アクション

2 ページの表 4 にリストされたパッケージのすべてがパスに含まれていることを確認してから、もう一度 **xlc_install** ユーティリティーを実行してく

ださい。 **xc_install** ユーティリティーをインストール・イメージで指定された以外の場所に移動した場合、**-rpmloc rpmlocation_path** オプションを使用する必要があります。詳しくは、13 ページの『**xc_install** オプション』を参照してください。

32 ビットまたは 64 ビット GCC (RHEL4U2、Y-HPC、YDL) の場所を判別できなかった

シナリオ

new_install または **vac_configure** ユーティリティーを実行して、RHEL4U2、Y-HPC、または YDL を実行中のコンピューターでコンパイラーを構成しようとしていたときに、以下のエラー・メッセージの少なくとも 1 つが受け取られました。

ERROR: Could not determine location of 32-bit GCC. Suggestion: Ensure 32-bit "glibc-devel", 32-bit "libstdc++-devel" are installed. These packages can be obtained from your operating system install media.

ERROR: Could not determine location of 64-bit GCC. Suggestion: Ensure 64-bit "glibc-devel", 64-bit "libstdc++-devel" are installed. These packages can be obtained from your operating system install media.

ERROR: Please ensure all relevant 32 and 64-bit GCC packages are installed before running "new_install" again. If they are installed but cannot be detected by "new_install", please run "**vac_configure**" manually.

説明 以下のパッケージのいずれかまたは両方が、適切なディレクトリーにインストールされていません。

- glibc-devel
- libstdc++-devel

注: パッケージを照会しても、RHEL4U2、Y-HPC、または YDL で、GCC の 32 ビットまたは 64 ビット・バージョンがインストールされているかどうかを判断できません。32 ビットのパッケージと 64 ビットのパッケージがともにまったく同じ名前を持っているからです。

アクション

テスト・ケースを 32 ビットと 64 ビットの両方のモードでコンパイルして、glibc-devel および libstdc++-devel の 32 ビットと 64 ビットの両方のパッケージがシステムにインストールされていることを確認してください。テスト・プログラムが正常にエラー・メッセージなしでコンパイルされれば、そのパッケージがインストールされていることになります。エラー・メッセージが出れば、そのパッケージをインストールする必要があることを意味します。

例 この例では、よく知られた「Hello World」テスト・ケースの事例を使用します。

1. GCC の 32 ビット・モード・バージョンのテストでは、以下のコマンドを実行します。

```
gcc helloWorld.c
g++ helloWorld.cpp
```

2. GCC の 64 ビット・モード・バージョンのテストは、以下のとおりです。

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

3. プログラムを正常にコンパイルできない場合は、必要なパッケージをインストールして、コンパイラーを再び構成する必要があることを意味します。
- a. 32 ビット・モード・テストでエラー・メッセージを受け取った場合は、以下のコマンドを使用して、必要なパッケージをインストールします。

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

ここで、*V.R.M-F* は、そのパッケージのバージョン.リリース.モディフィケーション-修正レベルを表します。

注: 32 ビット *glibc-devel* および *libstdc++-devel* パッケージは、オペレーティング・システムと共に提供されるインストール・メディアから入手可能です。パッケージ・ファイル名は、パッケージが 32 ビット・モード用か、または 64 ビット・モード用であることを示します。32 ビット・モードのパッケージ・ファイル名は、*.ppc.rpm です。

- b. 64 ビット・モード・テストでエラー・メッセージを受け取った場合は、以下のコマンドを使用して、必要なパッケージをインストールします。

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

ここで、*V.R.M-F* は、そのパッケージのバージョン.リリース.モディフィケーション-修正レベルを表します。

注: 64 ビット *glibc-devel* および *libstdc++-devel* パッケージは、オペレーティング・システムと共に提供されるインストール・メディアから入手可能です。パッケージ・ファイル名は、パッケージが 32 ビット・モード用か、または 64 ビット・モード用であることを示します。64 ビット・モードのパッケージ・ファイル名は、*.ppc64.rpm です。

- c. **new_install** または **vac_configure** を再び実行します。

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものであり、本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-8711
東京都港区六本木 3-2-12
IBM World Trade Asia Corporation
Intellectual Property Law & Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

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

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができませんが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確証できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

プログラミング・インターフェース情報

プログラミング・インターフェース情報は、プログラムを使用してアプリケーション・ソフトウェアを作成する際に役立ちます。

一般使用プログラミング・インターフェースにより、お客様はこのプログラム・ツール・サービスを含むアプリケーション・ソフトウェアを書くことができます。

ただし、この情報には、診断、修正、および調整情報が含まれている場合があります。診断、修正、調整情報は、お客様のアプリケーション・ソフトウェアのデバッグ支援のために提供されています。

注: 診断、修正、調整情報は、変更される場合がありますので、プログラミング・インターフェースとしては使用しないでください。

商標

以下は、IBM Corporation の商標です。

BladeCenter

IBM

IBM (ロゴ)

POWER

System i

System p

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

業界標準

以下の標準がサポートされています。

- C 言語は、International Standard for Information Systems-Programming Language C (ISO/IEC 9899-1999 (E)) と整合性があります。
- C++ 言語は、International Standard for Information Systems-Programming Language C++ (ISO/IEC 14882:1998) と整合性があります。
- C++ 言語は、International Standard for Information Systems-Programming Language C++ (ISO/IEC 14882:2003 (E)) と整合性があります。
- C および C++ 言語は、OpenMP C および C++ アプリケーション・プログラミング・インターフェース・バージョン 2.5 と整合性があります。

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

一時ファイル
インストール・ログ 37
高度な最適化レベル 6
ハード・ディスク・スペース所要量 6
一般ユーザー向けの説明 v
インストール
オプション 13
特殊 18
パッケージの存在場所 17, 18
複数バージョン 15
インストール CD 1
インストールの検査 9
インストール・イメージ 1
インストール・ユーティリティ
オプション 13
の使用 21
インストール・ログ 37

[カ行]

各国語サポート 2
基本インストールの定義 3
基本例の説明 ix
キャッシュ
ハード・ディスク・スペース所要量 6
共存
コンパイラー 15
更新
インストール・ユーティリティ・オプション 14
試行 17, 21
ステップ 21
前提条件 21
適用 21
デフォルト・ロケーション以外の場所への 21
パッケージ 21
構成ファイル
上書き 28
カスタマイズ vi, 25, 26
生成 13, 28
セキュリティ 35
デフォルト 28

構成ファイル (続き)
名前変更 28
バックアップ 25
複数 vi, 25
変更 vi, 25, 26
編集 26

[サ行]

使用可能なスペースの判別 8
上級者向けインストールの定義 4
上級者向けの説明 v
上級者向けの手順 15

[タ行]

デバッガー・サポート 13, 14
デバッグ 37

[ハ行]

ハード・ディスク・スペース、使用可能な 8
パッケージ
必要 6
見つからない 37
パッケージ、インストール 1
プリインストール 1
計画 3
ヘルプ
表示 13

[ラ行]

ロケール
サポートされる 2

G

gcc-c++, インストール済みバージョンの判別 9
GNU 6, 9

J

Java 6, 9

P

Perl 6, 9

R

Red Hat Package Manager (RPM)
インストール・イメージ 1
ユーティリティ 1

X

XL コンパイラー
共存 15



プログラム番号: 5724-M16

GD88-6731-01



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12