

XL C/C++ Advanced Edition for Linux®



インストール・ガイド

バージョン 7.0

XL C/C++ Advanced Edition for Linux®



インストール・ガイド

バージョン 7.0

ご注意

本書および本書で紹介する製品をご使用になる前に、21 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、XL C/C++ Advanced Edition for Linux のバージョン 7.0、および新しい版で特に明記されていない限り、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。

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： GC09-7946-00
XL C/C++ Advanced Edition for Linux®
Installation Guide
Version 7.0

発 行： 日本アイ・ビー・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 2004.8

この文書では、平成明朝体™W3、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™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 1997, 2004. All rights reserved.

© Copyright IBM Japan 2004

目次

インストール・ガイド 1

XL C/C++ のパッケージ化とファイルセット 1

XL C/C++ Advanced Edition for Linux バージョン

7.0 のインストール 2

システム前提条件 2

前提条件の作業と条件 2

インストール 4

RHEL3 への製品マイグレーション・インストール 7

インストール済みパッケージの照会 10

XL C/C++ マニュアル・ページの使用可能化 10

正しい NLSPATH の設定 11

コンパイラの構成 12

ドキュメンテーションの表示 14

呼び出しコマンドの環境の設定 15

インストールのテスト 15

XL C/C++ のアンインストール 16

トラブルシューティング 17

RHEL3 のトラブルシューティング 17

SLES9 のトラブルシューティング 18

RHEL3 および SLES9 の両方のトラブルシューティング 18

特記事項 21

商標 23

インストール・ガイド

本書には、IBM® XL C/C++ Advanced Edition for Linux のインストールに関する大切な情報が収められています。XL C/C++ Advanced Edition for Linux をインストールする前によくお読みください。また、CD-ROM に収められている **README** ファイルも併せてご参照ください。このファイルには XL C/C++ Advanced Edition for Linux の最新情報が収録されています。製品をインストールした後、*compiler-path/vacpp/7.0* ディレクトリーにある **README** ファイルを参照してください。

XL C/C++ のパッケージ化とファイルセット

以下のパッケージが CD に収録されています。rpm を使用して、これらのファイルセットを参照してください。

必要なパッケージ

パッケージ名	パッケージの説明
xlsmp.msg.rte	SMP ランタイム・メッセージ
xlsmp.rte	SMP ランタイム動的ライブラリー
xlsmp.lib	SMP ランタイム静的ライブラリー
xlmass.lib	IBM Mathematical Acceleration Subsystem (MASS) コンポーネント
vac.lic	XL C/C++ ライセンス
vacpp.rte	C++ ランタイム環境
vacpp.rte.lnk	C++ ランタイム環境のリンク
vac.lib	C コンパイラー・ライブラリー
vac.cmp	C コンパイラー
vacpp.lib	C++ コンパイラー・ライブラリー
vacpp.cmp	C++ コンパイラー
vacpp.samples	XL C/C++ サンプル
vacpp.help	XL C/C++ ドキュメンテーション

サポートされているロケールは次のとおりです。

- 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

XL C/C++ Advanced Edition V7.0 for Linux のインストール

IBM XL C/C++ Advanced Edition for Linux のインストールでは、Linux パッケージをインストールするための標準ツールである Red Hat Package Manager (RPM) を使用します。RPM はオペレーティング・システムに添付されています。

以下のハイレベルの手順を行う必要があります。

1. 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。
2. パッケージをインストールします。
3. コンパイラー・マニュアル・ページを使用可能にします。
4. 正しい NLSPATH を設定します。
5. コンパイラーを構成します。
6. 呼び出しコマンドの環境を設定します。
7. インストールをテストします。

システム前提条件

- **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 for POWER V3 (RHEL 3) アップデート 3
 - SUSE LINUX Enterprise Server 9 for POWER (SLES 9)
- **ハードウェア:** Linux ディストリビューションでサポートされる IBM pSeries™、iSeries™、または BladeCenter™ コンピューター。
- **ハード・ディスク・スペース所要量:**
 - 製品パッケージ用に約 100 MB。
 - 高レベルの最適化には、大量の追加ページング・スペースおよび一時ディスク・スペースが必要です。
- **必要なソフトウェア:**
 - HTML ブラウザー (ドキュメンテーションを表示するため)
 - PDF ビューアー (ドキュメンテーションを表示するため)

前提条件の作業と条件

XL C/C++ をインストールするには、root ユーザー・アクセスが必要です。

必要なパッケージの検査

コンパイラー・パッケージをインストールする前に、以下のパッケージをインストールしてください。

RHEL3	
GCC 前提条件	バージョン要件
gcc	3.2.3
gcc-c++	3.2.3
glibc-devel	2.3.2
libstdc++-devel	3.2.3

SLES9	
GCC 前提条件	バージョン要件
gcc	3.3.3
gcc-c++	3.3.3
gcc-64bit	9
glibc-devel-64bit	9
libstdc++-devel-64bit	9
java2	1.3.1
java2-jre	1.3.1

注: RHEL3 にインストールするには、32 ビットおよび 64 ビットの `glibc-devel` と `libstdc++-devel` のパッケージが両方とも必要です。コンパイラーをインストールする前に、これらの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、17 ページの『RHEL3 のトラブルシューティング』にある手順に従ってください。これらのパッケージがインストール済みであるかどうかを検査するのに、次のコマンドを使用することはできません。RHEL3 への RPM パッケージの名前は、RPM パッケージが 32 ビットか 64 ビットかに関係なく、同じ名前を持つように作成されているためです。結果として、出力では、32 ビットか 64 ビットか、あるいは両方の RPM パッケージがインストール済みであるかどうかを示されません。

パッケージがインストールされているかどうかを判別するには、次のコマンドを使用します。

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

例えば、`vac.cmp` がインストール済みであるかどうかを確認するには、`vac.cmp` パッケージを次のように照会します。

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

`vac.cmp` バージョン 7.0 がインストール済みであれば、照会の結果として次のような出力が得られます。

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

その他のパッケージの検査

次の項目にはその他のコンポーネントが必要です。また、そのものが他のコンポーネントの前提条件になっている場合もあります。コンパイラーを動作させるためには初期インストール後に **vac_configure** ユーティリティを実行しなければならぬので、このユーティリティが必要になります。コンパイラー資料にアクセスするための KDE デスクトップ環境とビューアーはオプションです。

その他のパッケージの依存関係

オプション?	コンポーネント	前提条件 RPM パッケージ	照会
必須	<code>vac_configure</code> ユーティリティ	<code>perl</code>	<code>rpm -qf /usr/bin/perl</code>
オプション	Mozilla または Konqueror ブラウザー	KDE 環境用 RPM パッケージ	<code>rpm -q mozilla</code> <code>rpm -q kdebase3-konqueror</code>
オプション	PDF 文書ビューアー	KDE 環境用 RPM パッケージ	<code>rpm -q xpdf</code>

スペース所要量の検査

XL C/C++ パッケージには、約 100 MB のハード・ディスク・スペースが必要です。この容量には、オプションのサンプルおよびドキュメンテーションが含まれます。高レベルの最適化でコンパイルすると、多くの一時ディスク・スペースが必要になります。

次のコマンドを使用して、デフォルトのインストール・ロケーションへのコンパイラのインストールに使用できるスペース量を判別してください。

```
df -h /opt
```

インストール

デフォルトでは、コンパイラは、**/opt/ibmcmp** ディレクトリーにインストールされます。

単一コマンドによる全パッケージのインストール

現行作業ディレクトリーに、XL C/C++ のすべての RPM パッケージが格納されていて、それ以外の RPM パッケージが入っていない場合は、次のように単一のコマンドを使用して XL C/C++ をインストールすることができます。

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

```
rpm -ivh *.rpm --prefix=installation-path (デフォルト以外のインストール)
```

以下の表に、RHEL と SLES9 の両方に必要なパッケージとその前提条件を示します。

XL C/C++ パッケージおよびその前提条件

パッケージ名	前提条件	説明	再配置
xlsmp.msg.rte	none	SMP ランタイム・メッセージ	同じロケーションにインストールすること
xlsmp.rte	xlsmp.msg.rte	SMP ランタイム動的ライブラリー	
xlsmp.lib	xlsmp.msg.rte xlsmp.rte	SMP ランタイム静的ライブラリー	
xlmass.lib	none	IBM Mathematical Acceleration Subsystem (MASS) コンポーネント	任意のロケーション
vac.lic	none	XL C/C++ ライセンス	任意のロケーション
vacpp.rte	none	C++ ランタイム環境	vacpp.rte および vacpp.rte.lnk を同じロケーションにインストールすること
vacpp.rte.lnk	vacpp.rte	C++ ランタイム環境のリンク	

XL C/C++ パッケージおよびその前提条件

パッケージ名	前提条件	説明	再配置
vac.cmp	vac.lib vac.lic xlmass.lib xlsmp.lib xlsmp.msg.rte xlsmp.rte	C コンパイラー	vac.cmp、vac.lib、vacpp.cmp、および vacpp.lib を同じロケーションにインストールすること
vac.lib	none	C コンパイラー・ライブラリー	
vacpp.cmp	vac.cmp vac.lic vac.lib vacpp.lib vacpp.rte vacpp.rte.lnk xlmass.lib xlsmp.lib xlsmp.msg.rte xlsmp.rte	C++ コンパイラー	
vacpp.lib	vac.lib	C++ コンパイラー・ライブラリー	
vacpp.samples	none	サンプル・プログラム	任意のロケーション (オプション)
vacpp.help	none	HTML および PDF 形式のコンパイラー資料	任意のロケーション (オプション)

RHEL3 へのデフォルト・インストール

XL C/C++ をデフォルトのロケーションである **/opt/ibmcmp** にインストールする場合は、以下のコマンドを実行します。インストール時に依存関係のエラーが発生しないように、コンパイラー・パッケージは以下の順序でインストールしてください。

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

サンプル・プログラムおよび製品資料パッケージには、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-7.0.0-0.ppc64pseries.rpm
rpm -ivh vacpp.samples-7.0.0-0.ppc64pseries.rpm
```

注: XL Fortran コンパイラーがすでにインストールされている場合は、**xlsmp.msg.rte**、**xlsmp.rte**、**xlsmp.lib**、および **xlmass.lib** はすでにインストール済みです。

ールされています。RPM では、これらのコンポーネントを 2 度インストールすることができないようになっています。

RHEL3 への非デフォルトのインストール

XL C/C++ をデフォルト以外のロケーションにインストールする場合は、以下のコマンドを実行します。インストール時に依存関係のエラーが発生しないように、コンパイラー・パッケージは以下の順序でインストールしてください。

```
rpm -ivh xlsmp.msg.rte-1.5.0-0.ppc64pseries.rpm --prefix=smp-rt-path
rpm -ivh xlsmp.rte-1.5.0-0.ppc64pseries.rpm --prefix=smp-rt-path
rpm -ivh xlsmp.lib-1.5.0-0.ppc64pseries.rpm --prefix=smp-rt-path
rpm -ivh xlmass.lib-4.1.0-0.ppc64pseries.rpm --prefix=xlmass-path
rpm -ivh vac.lic-7.0.0-0.ppc64pseries.rpm --prefix=vac-lic-path
rpm -ivh vac.lib-7.0.0-0.ppc64pseries.rpm --prefix=compiler-path
rpm -ivh vac.cmp-7.0.0-0.ppc64pseries.rpm --prefix=compiler-path
rpm -ivh vacpp.rte-7.0.0-0.ppc64pseries.rpm --prefix=vacpp-rt-path
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64pseries.rpm --prefix=vacpp-rt-path
rpm -ivh vacpp.lib-7.0.0-0.ppc64pseries.rpm --prefix=compiler-path
rpm -ivh vacpp.cmp-7.0.0-0.ppc64pseries.rpm --prefix=compiler-path
```

例:

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

SLES9 へのデフォルト・インストール

XL C/C++ をデフォルトのロケーションである **/opt/ibmcomp** にインストールする場合は、以下のコマンドを実行します。インストール時に依存関係のエラーが発生しないように、コンパイラー・パッケージは以下の順序でインストールしてください。

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

サンプル・プログラムおよび製品資料パッケージには、他の RPM パッケージとの依存関係がないため、任意の順序でインストールできます。

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

注: XL Fortran コンパイラーがすでにインストールされている場合は、

xlsmp.msg.rte、**xlsmp.rte**、および **xlsmp.lib** はすでにインストールされています。RPM では、これらのコンポーネントを 2 度インストールすることができないようになっています。

SLES9 への非デフォルトのインストール

XL C/C++ をデフォルト以外のロケーションにインストールする場合は、以下のコマンドを実行します。インストール時に依存関係のエラーが発生しないように、コンパイラー・パッケージは以下の順序でインストールしてください。

```
rpm -ivh xlsmp.msg.rte-1.5.0-0.ppc64.rpm --prefix=smprt-path
rpm -ivh xlsmp.rte-1.5.0-0.ppc64.rpm --prefix=smprt-path
rpm -ivh xlsmp.lib-1.5.0-0.ppc64.rpm --prefix=smprt-path
rpm -ivh xlmass.lib-4.1.0-0.ppc64.rpm --prefix=xlmass-path
rpm -ivh vac.lic-7.0.0-0.ppc64.rpm --prefix=vaclic-path
rpm -ivh vac.lib-7.0.0-0.ppc64.rpm --prefix=compiler-path
rpm -ivh vac.cmp-7.0.0-0.ppc64.rpm --prefix=compiler-path
rpm -ivh vacpp.rte-7.0.0-0.ppc64.rpm --prefix=vacpprt-path
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64.rpm --prefix=vacpprt-path
rpm -ivh vacpp.lib-7.0.0-0.ppc64.rpm --prefix=compiler-path
rpm -ivh vacpp.cmp-7.0.0-0.ppc64.rpm --prefix=compiler-path
```

例:

```
rpm -ivh xlsmp.msg.rte-1.5.0-0.ppc64.rpm --prefix=/common
rpm -ivh xlsmp.rte-1.5.0-0.ppc64.rpm --prefix=/common
rpm -ivh xlsmp.lib-1.5.0-0.ppc64.rpm --prefix=/common
rpm -ivh xlmass.lib-4.1.0-0.ppc64.rpm --prefix=common_xlmass
rpm -ivh vac.lic-7.0.0-0.ppc64.rpm --prefix=/license
rpm -ivh vac.lib-7.0.0-0.ppc64.rpm --prefix=/compiler
rpm -ivh vac.cmp-7.0.0-0.ppc64.rpm --prefix=/compiler
rpm -ivh vacpp.rte-7.0.0-0.ppc64.rpm --prefix=/runtime
rpm -ivh vacpp.rte.lnk-7.0.0-0.ppc64.rpm --prefix=/runtime
rpm -ivh vacpp.lib-7.0.0-0.ppc64.rpm --prefix=/compiler
rpm -ivh vacpp.cmp-7.0.0-0.ppc64.rpm --prefix=/compiler
```

RHEL3 への製品マイグレーション・インストール

注: XL C/C++ バージョン 7.0 は SLES9 でサポートされる最初のバージョンであるので、製品マイグレーション・インストールは SLES9 には適用されません。

製品マイグレーション・インストール

旧バージョンのコンパイラーがインストールされている場合は、インストールする前に、旧バージョンのコンパイラーをすべてアンインストールすることを推奨します。ただし、複数バージョンのコンパイラーをシステム上に存在させる場合は、このセクションで説明する方法の 1 つに従ってください。

方法 1: 異なるロケーションへのインストール: バージョン 7.0 をバージョン 6.0 とは異なるロケーションにインストールすることができます。

例えば、バージョン 6.0 コンパイラーを、デフォルト・ロケーションの **/opt/ibmcmp** にインストール済みである場合は、6 ページの『RHEL3 への非デフォルトのインストール』にある手順に従って、バージョン 7.0 パッケージを別のロケーションにインストールします。

バージョン 6.0 にデフォルト・インストールを使用せず、デフォルト以外のロケーションにインストールした場合は、バージョン 7.0 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できます。 5 ページの『RHEL3 へのデフォルト・インストール』にある手順に従ってください。同様に 6 ページの『RHEL3 への非デフォルトのインストール』にある手順に従って、バージョン 7.0 を非デフォルトのロケーションにインストールすることもできます。

方法 2: 同じロケーションへのコンパイラーのインストール: また、バージョン 7.0 パッケージを、バージョン 6.0 パッケージが存在している同じロケーションにインストールすることもできます。この場合は、バージョン 7.0 ランタイム・パッケージのみが、そのロケーション上にインストールされ、バージョン 6.0 コンパイラーは、バージョン 7.0 ランタイム共用ライブラリーを使用します。将来の依存関係エラーを避けるため、バージョン 7.0 パッケージを同じロケーションにインストールする前に、次のパッケージを除去する必要があります。

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

バージョン 6.0 のパッケージがすべてインストール済みである場合は、以下のコマンドを使用して、バージョン 7.0 パッケージを同じロケーションにインストールします。インストール時に依存関係のエラーが発生しないように、コンパイラー・パッケージは以下の順序でインストールしてください。

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

ここで、

<V1.3_smprt_path> は、旧バージョン 1.3 の smprt がインストール済みの場所です。

<V6.0_compiler_path> は、旧バージョン 6.0 コンパイラーがインストール済みである場所です。

<V6.0_license_path> は、旧バージョン 6.0 のコンパイラー・ライセンスがインストール済みである場所です。

<V6.0_runtime_path> は、旧バージョン 6.0 のコンパイラー・ランタイムがインストール済みである場所です。

<xlmass_path> にはシステム上の任意のロケーションを指定できます。

注:

1. ファイル競合を避けるため、ランタイム・パッケージ上で **--force** オプションを使用する必要があります。 **--force** オプションを使用しないと、次のような警告が表示されます。

```
file /opt/ibmcmp/lib/libibmc++.so.1 from install of vacpp.rte-7.0.0-0
conflicts with file from package vacpp.rte-6.0.1-0
```


2. マイグレーション・インストールのために方法 2 を使用する場合は、手動で **vac_configure** を使用する必要があります。システム上に複数バージョンのコンパイラーがインストールされている場合は、**new_install** を使用しないでください。
3. 方法 2 を使用する場合、バージョン 7.0 ランタイム・パッケージは、両方のバージョンのコンパイラーによって使用されます。バージョン 6.0 ランタイム・パッケージを、バージョン 6.0 ランタイム・パッケージを使用して実行する必要がある場合は、方法 1 を使用しなければなりません。

PTF 更新

PTF (プログラム一時修正) 更新の場合は、**vac.lic** を除くすべてのパッケージが同梱されています。

PTF 更新を適用する前に、旧バージョンのコンパイラーをすべてアンインストールすることを推奨します。この場合は、旧バージョンから存在している **vac.lic** を使用する必要があります。しかし、PTF 更新を適用する場合、設定に基づいて、旧バージョンのコンパイラーを保持するかアンインストールするかを選択できます。

方法 1: 旧バージョンのコンパイラーの保持および PTF 更新の適用: 旧バージョンのコンパイラーをシステム上に保持する場合は、PTF を別のロケーションにインストールする必要があります。この場合、**vac.lic** を再インストールする必要はありませんが、旧バージョンのコンパイラーをアンインストールしない場合は、**vac_configure** オプションを使用する必要があるということに注意してください。**new_install** は使用しないでください。

後で、PTF が必要であることを確認し、旧バージョンをアンインストールすることにした場合は、**vac.lic** を除くすべてをアンインストールできます。旧バージョンのコンパイラーをシステムからアンインストールしないで、同じインストール・ロケーションに PTF を適用しようとする、インストールが失敗します。これは、製品のバージョン専用ではない共通ロケーションにパッケージがインストールされたためです。以下のステップに従うと、PTF を適用することができます。

1. 4 ページの『インストール』で説明しているように新しい RPM パッケージを使用して、手順を完了する。**vac.lic** パッケージはすでにインストール済みであるので、再度インストールする必要はありません。
2. **vac_configure** を使用してコンパイラーを構成する。**new_install** ユーティリティを使用しないでください。
3. PTF 更新を適用するとき、旧バージョンのコンパイラーを保持することにし、後で、旧コンパイラーをアンインストールしなくなった場合は、16 ページの『XL C/C++ のアンインストール』にある手順に従う。**vac.lic** はアンインストールしないでください。このパッケージをアンインストールしようとする、エラー・メッセージが出されます。これは、このパッケージが、ご使用のシステムにインストール済みのほかのパッケージに必要な前提条件であるためです。

方法 2: 旧バージョンのコンパイラーのアンインストールおよび PTF 更新の適用: PTF 更新を適用するとき、旧バージョンのコンパイラーを望まない場合は、16 ページの『XL C/C++ のアンインストール』で説明されている手順に従い、4 ページの『インストール』にある手順を使用して PTF 更新を適用します。旧バージョンから存在している **vac.lic** を使用する必要があります。

インストール済みパッケージの照会

個々のパッケージを照会するには、以下のようなコマンドを実行します。

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

インストールが正常に終了しなかった場合は、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が出されます。パッケージが正しくインストールされた場合、結果は次のようになります。

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

ここで、*V.R.M-F* は、システムにインストールされたコンパイラーの Version.Release.Modification.Fix レベルのことです。

すべてのパッケージを照会し、*vac* を検索するには、次のコマンドを実行します。

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

パッケージがインストールされていない場合、コマンドの出力はヌルとなります。

XL C/C++ マニュアル・ページの使用可能化

コンパイラー起動コマンド、およびコンパイラーに付属のその他のユーティリティーについてのマニュアル・ページが用意されています。

マニュアル・ページを参照できるようにするには、事前に以下のように **MANPATH** 環境変数に **XL C/C++ man** ディレクトリーを追加しておかなければなりません。

1. 端末ウィンドウを開きます。
2. 次のコマンドを入力します。

```
export LANG=locale
```

マニュアル・ページでサポートされているロケールは次のとおりです。

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

例えば、英語のマニュアル・ページを表示するには、次のコマンドを入力します。

```
export LANG=en_US
```

3. 次の表に示すように、**MANPATH** 環境変数の先頭に **/opt/ibmcmp/vacpp/7.0/man** を追加します。

MANPATH 環境変数の更新方法

シェル	コマンド
bash	export MANPATH=/opt/ibmcmp/vacpp/7.0/man:\$MANPATH

マニュアル・ページを呼び出すには、例えば以下のように、**man** の後にコマンドを入力します。

```
man xlc.
```


マニュアル・ページを終了するには、**q** と入力します。

正しい NLSPATH の設定

インストールの後に、NLSPATH 環境変数を設定し、ランタイム環境が適切なメッセージ・カタログを検出するようにします。

次のコマンドを実行してください:

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

ここで、

- *smprrt-path* は SMP ランタイムのインストール・ロケーションです。
- *vacpprrt-path* は C++ ランタイム環境のインストール・ロケーションです。
- *compiler-path* は C および C++ コンパイラーのインストール・ロケーションです。

注: デフォルトのインストール・ロケーションが使用されている場合、*smprrt-path*、*vacpprrt-path*、および *compiler-path* は、すべて **/opt/ibmcomp** です。

コンパイラーの構成

new_install を実行する場合は、コンパイラーを構成するため、root 特権または管理者特権がなければなりません。ただし、コンパイラーを、**vac_configure** を使用して構成する場合で、出力構成ファイル *vac.cfg* が、書き込み許可のあるロケーションに書き込まれる場合は、root 特権も管理特権も必要ありません。

new_install を使用して、デフォルト構成ファイルを自動的に生成することができます。**new_install** コマンドは、RPM データベースを照会して、XL C/C++ パッケージへのパスを検索し、さらに 32 ビット GCC (*gcc32path*) および 64 ビット GCC (*gcc64path*) へのパスを PATH 環境変数の中で検索して、**/etc/opt/ibmcomp/vac/7.0/vac.cfg** 構成ファイルを生成します。このコマンドは、得られた値を使用して、**vac_configure** を実行します。SLES9 プラットフォーム上で **new_install** を実行すると、構成ファイルが生成される前に、ご使用条件を受け入れることを要求されます。このステップは、RHEL3 では要求されません。

1. **new_install** および **vac_configure** の実行可能ファイルが入っているディレクトリに変更します。

```
cd /opt/ibmcomp/vacpp/7.0/bin
```

または、XL C/C++ をデフォルト以外のロケーションにインストールした場合は、以下のディレクトリに変更します。

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

2. 以下のコマンドを実行します。

```
./new_install
```

new_install コマンドは以下のコマンドを実行します。

```
vac_configure  
-gcc gcc32path  
-gcc64 gcc64path
```

```

-install
-mass xlmass-path/xlmass/4.1
-smprt smprt-path/xlsmp/1.5
-vaclic vaclic-path/vac/7.0
-vacpprt vacpprt-path/vacpp/7.0
-vac compiler-path/vac/7.0
-vacpp compiler-path/vacpp/7.0
compiler-path/vac/7.0/etc/vac.base.cfg

```

new_install を起動する前に、デフォルトの構成ファイルが存在している場合は、**new_install** が既存の構成ファイルをバックアップします。バックアップ・ファイルの名前が画面に表示されます。

注: XL C/C++ または GCC の複数インスタンスがシステムにインストールされている場合は、**vac_configure** を手動で使用する必要があります。

new_install がエラーで終了した場合は、『vac_configure ユーティリティ』の説明に従って、**vac_configure** ユーティリティを手動で実行する必要があります。

以下の表に、生成された構成ファイル内の属性の一部を示します。

Linux 固有の構成属性

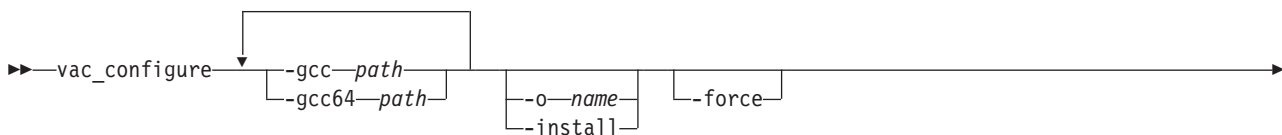
属性	内容	追加情報
gcc_path gcc_path_64	32 ビット・モードまたは 64 ビット・モードの GCC 実行可能ファイルのロケーション	gcc コマンドは、指定されたパス下の bin ディレクトリーに入っていないとなりません。
gcc_libs gcc_libs_64	32 ビット・モードまたは 64 ビット・モードの、GCC ライブラリーがコンマで区切られたリスト	gcc は、このリストを vac_configure ユーティリティに戻します。
gcc_libdirs gcc_libdirs_64	32 ビット・モードまたは 64 ビット・モードの、GCC ライブラリーが入っているディレクトリーがコンマで区切られたリスト	gcc は、このリストを vac_configure ユーティリティに戻します。

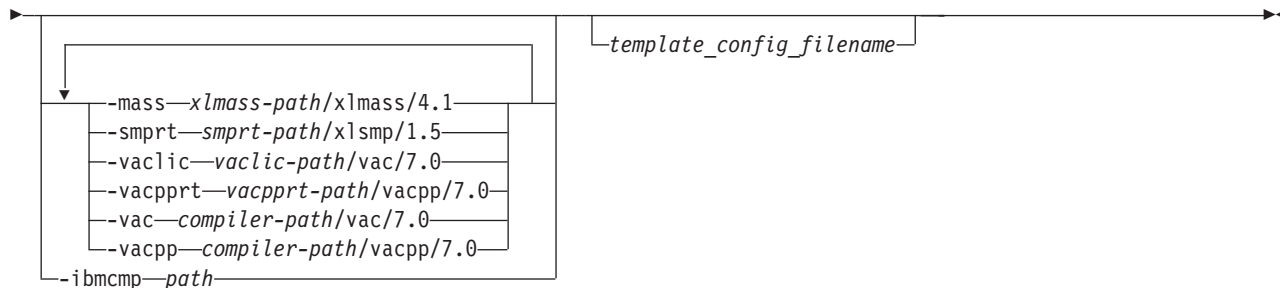
vac_configure ユーティリティ

vac_configure ユーティリティは、**new_install** コマンドによって自動的に起動されます。以下の場合、ユーザーが **vac_configure** ユーティリティを直接起動することができます。

- **new_install** コマンドからエラーを受け取った場合。
- **new_install** の実行は正常終了したが、後でデフォルトの構成ファイルを変更したくなった場合 (GCC の新しいバージョンをインストールする場合など)。
- 追加構成ファイルを作成する場合。

vac_configure コマンドには、以下の構文があります:





ここで、

-gcc path

GCC bin ディレクトリーをインストールするパスを指定します。構成ファイルでは、`gcc_path` 属性はこのパスと同じものに設定されます。

例えば、GCC コマンドが `/usr/bin/gcc` にある場合は、次のように指定します。

```
-gcc /usr
```

-gcc64 path

64 ビットの GCC bin ディレクトリーをインストールするパスを指定します。構成ファイルでは、`gcc_path_64` 属性はこのパスと同じものに設定されます。

-o name

生成する構成ファイルの名前を指定します。デフォルトでは、出力がディスプレイへ書き込まれます。

-install

デフォルトの構成ファイルの必要名と必要なロケーションでファイルを生成します: `/etc/opt/ibmcmp/vac/7.0/vac.cfg`

-force vac_configure ユーティリティーで、指定の名前およびパスを使用して既存の出力ファイルを強制的に上書きします。デフォルトでは、`force` を使用しないで、指定のファイルがすでに存在する場合は、**vac_configure** はエラー・メッセージを出して、停止します。

-mass xlmass-path/xlmass/4.1

xlmass.lib パッケージのパスを指定します。デフォルトでは、このパスは `/opt/ibmcmp/xlmass/4.1` です。

-smprt smprt-path/xlsmp/1.5

xlsmp.msg.rte、**xlsmp.rte**、および **xlsmp.lib** パッケージのパスを指定します。デフォルトでは、このパスは `/opt/ibmcmp/xlsmp/1.5` です。

-vaclic vaclic-path/vac/7.0

vac.lic パッケージのパスを指定します。デフォルトでは、これは `/opt/ibmcmp/vac/7.0` です。

-vacpprt vacpprt-path/vacpp/7.0

vacpp.rte および **vacpp.rte.lnk** パッケージのパスを指定します。デフォルトでは、これは `/opt/ibmcmp/vacpp/7.0` です。

-vacpp *compiler-path/vacpp/7.0*
vacpp.cmp パッケージのパスを指定します。デフォルトでは、これは **/opt/ibmcmp/vacpp/7.0** です。

-vac *compiler-path/vac/7.0*
vac.cmp パッケージのパスを指定します。デフォルトでは、これは **/opt/ibmcmp/vac/7.0** です。

-ibmcmp *path*
別の方法として、すべての XL C/C++ パッケージ (**xlsmp.msg.rte**、**xlsmp.rte**、**xlsmp.lib**、**vac.lic**、**vacpp.rte**、**vacpp.rte.lnk**、**vac.cmp**、および **vacpp.cmp**) をインストールするパスを指定します。この場合、パスは、*smprrt-path*、*vaclic-path*、*runtime-path*、および *compiler-path* と同じです。 **ibmcmp** を指定して、さらに個々のパスを指定することはできません。

template_config_filename
構成ファイルを作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/7.0/etc/vac.base.cfg** です。 **vac.cmp** パッケージを再配置して、デフォルトのテンプレートを使用しない場合は、以下を指定します:

compiler-path/vac/7.0/etc/vac.base.cfg.

ドキュメンテーションの表示

XL C/C++ には、以下のドキュメンテーションが用意されています。

README ファイル

README ファイルは、インストール CD のルート・ディレクトリに入っています。このファイルは、**/opt/ibmcmp/vacpp/7.0** ディレクトリにインストールされます。

PDF ブック XL C/C++ ドキュメンテーションの PDF バージョンは、インストール CD の **/doc/\$LANG/pdf** ディレクトリに入っています。
vacpp.help パッケージをインストールすると、PDF ファイルは **/opt/ibmcmp/vacpp/7.0/doc/\$LANG/pdf** ディレクトリにコピーされます。

HTML ファイル

vacpp.help パッケージをインストールすると、XL C/C++ ドキュメンテーションの HTML バージョンは **/opt/ibmcmp/vacpp/7.0/doc/\$LANG/html** ディレクトリにコピーされます。HTML ファイルを表示するには、Mozilla、Konqueror などのブラウザでファイル **index.htm** を開きます。

マニュアル・ページ

コンパイラ起動コマンド (**xlc**) および追加コマンド (**vac_configure**、**new_install**、**gxlc**、**gxlc++**、**showpdf**、**mergepdf**、**resetpdf**、および **cleanpdf**) のマニュアル・ページが用意されています。マニュアル・ページは、**/opt/ibmcmp/vacpp/7.0/man/\$LANG/man1** ディレクトリにインストールされます。

呼び出しコマンドの環境の設定

XL C/C++ は **/usr/bin** に自動的にインストールされません。絶対パスを指定せずにコンパイラーを呼び出すには、以下のステップのうちの 1 つを行なってください。

- 以下のように、**/opt/ibmcmp/vac/7.0/bin** および **/opt/ibmcmp/vacpp/7.0/bin** に含まれている特定のドライバー用の **/usr/bin** へのシンボリック・リンクを作成します。

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

例:

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

- PATH 環境変数に **/opt/ibmcmp/vac/7.0/bin** および **/opt/ibmcmp/vacpp/7.0/bin** を追加します。

インストールのテスト

製品インストールおよびクリティカルな検索パスをテストするには、以下の単純な C および C++ プログラムを作成してみてください。

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. **xlc** コマンドを使用してテスト・プログラムをコンパイルします。例:

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

3. 次のプログラムを実行します。

```
./hello
```

画面に "Hello World!" と出力されるはずです。

4. プログラムの終了コードを調べます。

```
echo $?
```

結果はゼロになるはずです。

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.c++** にします。

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

6. **xlc++** コマンドを使用してテスト・プログラムをコンパイルします。例:

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

7. 次のプログラムを実行します。

```
./hello
```

画面に "Hello World!" と出力されるはずです。

8. プログラムの終了コードを調べます。

```
echo $?
```

結果はゼロになるはずです。

XL C/C++ のアンインストール

このプロダクト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。パッケージをアンインストールする順序は、それらをインストールした順序と逆、つまり LIFO (後入れ先出し法) であることに注意してください。アンインストール時に、依存関係エラーを避けるため、コンパイラー・パッケージを次の順序でアンインストールしてください (デフォルトおよび非デフォルトのインストール・ロケーションは両方とも、同じセットのコマンドを使用します)。

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

注:

1. 他のパッケージがパッケージの一部を必要とする場合、そのパッケージをアンインストールできない可能性があります。例えば、SMP ランタイム (xlsmp.rte) は、IBM XL Fortran も同じシステムにインストールされている場合は、共用コンポーネントであると考えられます。
2. 上記のコマンドは、**new_install** または **vac_configure** によって生成された vac.cfg を除去しません。

サンプル・プログラムと製品資料には、パッケージの依存関係がありません。これらをアンインストールするには、次のコマンドを任意の順序で発行します。

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

トラブルシューティング

このセクションでは、XL C/C++ Advanced Edition for Linux のインストール時に生じる可能性のある問題について説明します。

RHEL3 のトラブルシューティング

製品を正常にインストールして、構成するには、ppc および ppc64 の glibc-devel と libstdc++-devel のパッケージが両方ともなければなりません。 **new_install** または **vac_configure** を実行するとき、 ppc RPM パッケージだけがインストール済みで、 ppc64 RPM パッケージがインストールされていない場合は、以下のいずれかのエラー・メッセージが表示されます。

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

この場合は、glibc-devel および libstdc++-devel の、 64 ビットと 32 ビットの RPM パッケージが両方ともシステムにインストール済みであるかどうかを検査する必要があります。 RHEL3 の場合、 32 ビットまたは 64 ビット・バージョンの GCC がインストール済みであるかどうかは、パッケージの照会ではわかりません。これは、32 ビットおよび 64 ビット・パッケージがまったく同じ RPM 名を持っているためです。しかし、32 ビットおよび 64 ビット・モードでテスト・ケースをコンパイルすることによって、これを検査することができます。

例えば、 32 ビットまたは 64 ビット・バージョンの GCC がインストール済みであるかどうかを検査するには、 2 つのテスト・ケース、helloWorld.c と helloWorld.cpp を作成して、以下のようにファイルのコンパイルを試行することができます。

32 ビット・モードの場合:

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

プログラムが、エラー・メッセージなしで正常にコンパイルした場合は、 32 ビットの RPM パッケージがインストール済みであることを示しています。そうではなく、エラー・メッセージが表示された場合は、 32 ビットの RPM パッケージがインストール済みではないことを意味するので、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する必要があります。

```
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、システムにインストールされたパッケージの Version.Release.Modification.Fix レベルのことです。

例えば、以下のコマンドを使用することができます。


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

64 ビット・モードの場合:

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

プログラムが、エラー・メッセージなしで正常にコンパイルした場合は、64 ビットの RPM パッケージがインストール済みであることを示しています。そうではなく、エラー・メッセージが表示された場合は、64 ビットの RPM パッケージがインストール済みではないことを意味するので、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する必要があります。

```
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、システムにインストールされたパッケージの Version.Release.Modification.Fix レベルのことです。

例えば、以下のコマンドを使用することができます。

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

パッケージをインストールした後、**new_install** を再度実行すると、製品が、エラーなしでインストールされるはずです。

SLES9 のトラブルシューティング

Java2 パッケージは、**new_install** を実行する必要があります。ご使用のマシンに java2 または java2-jre がインストール済みではない場合に、**new_install** を実行すると、以下のエラー・メッセージが出されます。

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

オペレーティング・システム・インストール・メディアから得られる java2 および java2-jre パッケージをインストールして、**new_install** を再度実行する必要があります。

RHEL3 および SLES9 の両方のトラブルシューティング

- **new_install** または **vac_configure** の実行で問題が生じた場合は、以下のオプションを使用してさらに詳細な出力を得ることができます。
 - v (詳細) 一部のデバッグ情報を提供します。
 - vv (非常に詳細) -v よりもさらに詳細なデバッグ情報を提供します。
- **new_install** を適切に実行するには、gcc および gcc-c++ の両方の RPM パッケージがインストール済みで、PATH 環境変数がそれらのパスを検出できるようになっていなければなりません。そうでない場合は、**new_install** を実行すると、次のエラー・メッセージが出される可能性があります。

Could not determine location of gcc

または

Could not determine location of g++

PATH 環境変数内で gcc および g++ が検出されるのを望まない場合は、`-gcc` および `-gcc64` を手動で実行し、好みの gcc パスを使用して **vac_configure** を実行することができます。gcc および g++ を PATH から検出できるかどうかを判別するには、次のコマンドを使用できます。

```
which gcc
```

例えば、上記のコマンドを実行すると、次のような出力が得られます。

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

この出力は、gcc を、PATH 環境変数にリストされているロケーションの 1 つで検出できることを示しています。出力がヌルである場合、gcc/g++ へのパスを知っていれば、**vac_configure** を手動で実行する必要があります。

- XL Fortran コンパイラーがすでにインストールされている場合は、`xlsmplib.rte`、`xlsmplib`、および `xlmasslib` パッケージがすでにインストール済みであり、RPM は、これらのパッケージが再度インストールされるのを防止します。例えば、XL Fortran がインストール済みである場合に、次のコマンドを使用して、`xlsmplib` および `xlmasslib` パッケージをインストールしようとすると、

```
rpm -ivh xlsmplib.rpm xlmasslib.rpm
```

次のようなエラー・メッセージが出されます。

```
package xlsmplib-1.5.0-0 is already installed
package xlsmplib.rte-1.5.0-0 is already installed
package xlsmplib.rte-1.5.0-0 is already installed
package xlmasslib-4.1.0-0 is already installed
```

このメッセージは予想されたものであり、解決のために何かをする必要はありません。

- パッケージを、その前提条件をインストールしないでインストールしようとすると、そのパッケージはインストールされず、エラー・メッセージが出されます。例えば、`xlsmplib` パッケージの前に、`vac.cmp` をインストールしようとすると、`vac.cmp` はインストールされず、エラー・メッセージが、必要なパッケージを前提条件として示します。`vac.cmp` をインストールするために、次のコマンドを実行します。

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

`xlsmplib` パッケージが存在していなければ、次のようなエラー・メッセージが出されます。

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

前提条件を最初にインストールして、パッケージを再度インストールする必要があります。インストール時に依存関係のエラーが発生しないように、この資料のデフォルト・インストールまたは非デフォルト・インストールのセクションで述べた正しい順序でコンパイラー・パッケージをインストールしてください。

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。

〒106-0032
東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
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
B3/KB7/8200/MKM
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すが、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にお願いします。

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。お客様は、IBM 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。

商標

以下は、IBM Corporation の商標です。

- IBM
- iSeries
- pSeries

他の会社名、製品名およびサービス名などはそれぞれ各社の商標または登録商標です。



プログラム番号: 5724-K77

GC88-9979-00



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