

Microsoft INTERIX Release 2.2

Getting Started

Microsoft Corporation

One Microsoft Way

Redmond, Washington

USA 98052-6399

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

Microsoft INTERIX Release 2.2 Getting Started

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, people and events depicted herein are fictitious and no association with any real company, organization, product, person or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 1999-2000 Microsoft Corporation. All rights reserved.

Microsoft, MS-DOS, Windows, Windows NT, and Microsoft Visual C/C++ are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries/regions.

Portions of this software distribution are derived from code owned by the Regents of the University of California. Copyright © 1989, 1993 The Regents of the University of California. All rights reserved.

Portions of this software distribution are derived from Free Software Foundation, Inc. source code. As required by the FSF license, changes to the source code are available from the INTERIX web site's Tool Warehouse. The URL is <http://www.microsoft.com/windows/sfu/>.

Adobe and Acrobat are trademarks of Adobe Systems Incorporated which may be registered in certain jurisdictions.

Pentium and Pentium Pro are registered trademarks of Intel Corporation.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company, Ltd.

Open Software Foundation, OSF, OSF/Motif, and Motif are trademarks of The Open Software Foundation, Inc.

X Window System is a trademark of X Consortium, Inc.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Table of Contents

Introduction	5
Welcome	5
OPENNT to Microsoft INTERIX	5
Microsoft INTERIX 2.2 Components.....	6
Microsoft INTERIX 2.2	6
Microsoft INTERIX 2.2 Software Development Kit	7
Microsoft INTERIX PDF Documentation	7
Hardware/Software Requirements	7
Contacting Technical Support.....	8
Installation	9
Before You Install.....	9
Microsoft INTERIX 2.2.....	10
Microsoft INTERIX Software Development Kit	10
Saving Disk Space	11
Uninstalling Microsoft INTERIX	11
Microsoft INTERIX Basics	12
Microsoft INTERIX PDF Documentation	12
Microsoft INTERIX 2.2.....	13
Services: The Telnet Daemon and Syslogd	13
Working With Tape Drives.....	14
The Microsoft INTERIX X11 Clients	14
Window Managers and Window Modes.....	14
Microsoft INTERIX X11 Clients.....	15
Microsoft INTERIX X Clients on the Start Menu	15
Microsoft INTERIX Software Development Kit	16
Building an X11 Application	16

WELCOME

MICROSOFT INTERIX COMPONENTS

OPENNT TO MICROSOFT INTERIX

HARDWARE/SOFTWARE REQUIREMENTS

CONTACTING TECHNICAL SUPPORT

Introduction

Welcome

Thank you for purchasing Microsoft INTERIX, Microsoft's system environment for Windows NT that provides the most-requested functionality of UNIX systems.

This *Getting Started* booklet provides installation information and a quick introduction for the two Microsoft INTERIX components: the basic commands and utilities and the Microsoft INTERIX Software Development Kit.

Microsoft INTERIX gives you all the features of a traditional UNIX operating system, including pipes, hard file links, UNIX networking, and UNIX graphical support through the X Window System. It also provides UNIX and POSIX.2 utilities such as KornShell, *awk*, and *vi*. The Microsoft INTERIX environment is not an emulation; it is a native environment subsystem that integrates with the Windows NT kernel, just as the Win32 subsystem does. Shell scripts and other scripted applications that use UNIX and POSIX.2 utilities will run under Microsoft INTERIX. To port C source code from a UNIX system to Microsoft INTERIX, you will need the Microsoft INTERIX Software Development Kit.

Microsoft INTERIX Release 2.2 functionality includes *gcc* and *g++*, new libraries and utilities, *syslogd* and *utmpx* functions, performance improvements, and improved interoperability with other filesystems such as NFS and Samba.

OPENNT to Microsoft INTERIX

Earlier releases of Microsoft INTERIX before release 2.2 went under the name OPENNT. Throughout the documentation, you'll still see references to OPENNT. To preserve backwards compatibility throughout the 2.x releases, the OPENNT pathnames, environment variables, and compiler macros stay the same in this release. For example, the name of the installation directory is still available through the environment variable *OPENNT_ROOT*.

If you are installing over a previous release, the new version is installed in the same directory. If you are installing Microsoft INTERIX for the first time, it is installed in the directory **C:\Interix**. Future releases will migrate to new names (such as renaming *OPENNT_ROOT* to be *INTERIX_ROOT*), but the old ones will remain for backwards compatibility during the 2.x series of releases.

Microsoft INTERIX 2.2 Components

This section describes the Microsoft INTERIX Release 2.2 package and the Microsoft INTERIX Release 2.2 products.

Your Microsoft INTERIX package consists of a CD-ROM with *all* of the product components for Microsoft INTERIX and the Microsoft INTERIX Software Development Kit.

Microsoft INTERIX 2.2

The basic Microsoft INTERIX package consists of the Microsoft INTERIX subsystem and over 290 utilities including X11R5 client applications and a multi-user telnet server. The Microsoft INTERIX Release 2.2 telnet server is a high performance, pseudo-terminal based, true UNIX-style telnet daemon.

This package does not include an X11 server. To use the X11R5 client applications, you will need an X11 server for your local display or an X11 server on a remote computer.

Features include:

- KornShell and C Shell
- BSD sockets and UNIX networking
- Memory mapped files
- SVID IPC mechanisms: message queues, semaphores and shared memory
- Pseudo-terminal support
- Color curses support
- Full integration with Windows NT security model
- Full integration with Windows NT file systems
- Over 290 utilities, including scripting tools such as *awk*, *sed*, and *Perl5*
- POSIX.2 shell and utilities
- X11R5 runtime including X clients such as *xterm*, *twm*, *xrdb* and *xlsclients*
- Support for running X Windows applications on remote displays
- Support for UNIX-style daemons running as Windows NT services
- *cron* job scheduler
- A multi-user *telnetd* daemon supports up to 25 simultaneous users logged into the system. You must have the appropriate client license pack for Windows NT Server to comply with licensing requirements.

The number of users logged in at any time depends upon the number of available pseudo terminals. (Other programs, such as *xterm* may use Pseudo terminals.) Microsoft INTERIX supports 255 pseudo-terminal connections.

Note that, although the Microsoft INTERIX telnet server will allow multiple users, you must have the appropriate Windows NT or Windows 2000 client access licenses.

Microsoft INTERIX 2.2 Software Development Kit

The Microsoft INTERIX Software Development Kit provides the tools and libraries needed to port UNIX applications to Microsoft INTERIX, or to develop new UNIX applications.

Features include:

- The GNU C and C++ compilers, *gcc* and *g++*
- *cc* and *c89* compiler front ends to work with the Microsoft Visual C/C++ compiler
- Development tools, including *make*, *RCS*, *lex*, *yacc*, *nm*, *ar*, *strip*
- ANSI C and POSIX.1 interfaces
- Color curses library
- BSD-style sockets library support
- X11R5 libraries and header files
- Math library **libm.a**

The Microsoft INTERIX Software Development Kit will only work if Microsoft INTERIX 2.2 (the subsystem, commands, and utilities) is already installed.

Microsoft INTERIX PDF Documentation

All of the packages contain on-line documentation in Adobe's Portable Document Format (PDF). These documentation files are selected and installed separately. Select the Microsoft INTERIX *Online Library* in the Microsoft INTERIX program group for a listing of available documents. The PDF reader is not provided in this package, but is available on the World Wide Web at <http://www.adobe.com>.

Hardware/Software Requirements

The following hardware is required to run Microsoft INTERIX:

- An Intel 486, Pentium or Pentium Pro (or compatible) computer
- Both uniprocessor and multiprocessor systems are supported
- A Windows NT-compatible CD-ROM drive, display, and mouse

The following software is required:

- Microsoft Windows NT 3.51 Service Pack 5 or NT 4.0 Service Pack 3 or above, Workstation or Server or Windows 2000. The service packs introduce essential functionality to Windows NT.

The components that make up the packages require the following hard disk space; note that packages require more disk space on a FAT file system than on an NTFS file system. Figures are rounded up to the nearest megabyte.

Microsoft INTERIX 2.2	44 megabytes
Microsoft INTERIX 2.2 SDK	50 megabytes

Contacting Technical Support

Thank you for using Microsoft INTERIX Products. Please use the email alias **interix@microsoft.com** for support regarding this product. If you are a Microsoft Premier or Alliance customer, please use the contact information included with your contract for support

BEFORE YOU INSTALL

MICROSOFT INTERIX 2.2

MICROSOFT INTERIX SOFTWARE DEVELOPMENT KIT

SAVING DISK SPACE

UNINSTALLING MICROSOFT INTERIX

Installation

Installing a Microsoft INTERIX product takes less than ten minutes with a CD-ROM.

The most recent information about the install process and about the product is in the release notes. The release notes are on the INTERIX CD-ROM in the file **\\READ1ST.TXT**.

Before You Install

Before running the setup program:

- Turn off (stop) any INTERIX services (such as the *telnetd* service) you have running, using either the *service* command or the Services applet of the Control Panel.
- If you are installing from a remote drive, the drive must be mounted as a drive letter. If you give a UNC name, the install program will report it cannot find the archive.
- Reboot the system. This ensures that a Microsoft INTERIX or POSIX subsystem is not running. (If any Microsoft INTERIX or POSIX program has been run since the last time the system was started, the subsystem **PSXSS.EXE** will be running. Rebooting a Windows NT system shuts down the subsystem.)
- Login to the Administrator account.
- Microsoft INTERIX has full POSIX file semantics only on NTFS partitions. It can be installed or used on a FAT partition, but features such as file links, file groups and ownerships and the full scope of file names allowed under POSIX are available only when you are working on an NTFS partition. If you do install the products on a FAT file system, you will still get full POSIX file semantics for files stored on NTFS file systems.

Follow the instructions given by the setup program. You must choose the type of installation and the root of the directory hierarchy for installation. The default installation location is the directory **C:\Interix**. The installation location is referred to in the rest of the documentation as **\$OPENNT_ROOT** or **\$INTERIX_ROOT**.

Microsoft INTERIX 2.2

To install the Microsoft INTERIX 2.2 product:

1. Using File Manager, Explorer, or a **CMD.EXE** console window, go to the directory **\X86** on the Microsoft INTERIX CD-ROM.
2. Run **SETUP.EXE**.
3. Follow the instructions given by the setup program. Interix no longer requires activation key input. Both Server and Software Development Kit packages will be installed.
4. Select the installation directory on a local drive. The default is **C:\Interix**

Note: Do *not* install into the **Program Files** subdirectory or any other directory name containing a space. The Microsoft INTERIX utilities will have difficulties with the space in the directory name. To install into a directory whose name contains spaces, supply the “short” MS-DOS name for the directory; for example, enter **PROGRA~1** for **PROGRAM FILES**. (The MS-DOS name is visible under the directory’s properties in Explorer or by selecting View->All File Details in File Manager.)

5. Select the components you want to install.

Once installation is complete, you must logout and login again. The new registry and environment entries will not take effect until you logout and login again.

If you require a conforming POSIX environment, remove “Bypass Traverse Checking” permission for all users (that is, for user “Everyone”). In the On-Line Library, see chapter 3 of the *User’s Guide*, “The INTERIX Environment.”

Microsoft INTERIX Software Development Kit

The Microsoft INTERIX Software Development Kit requires Microsoft INTERIX 2.2. Normally the Microsoft INTERIX SDK is installed at the same time as the Microsoft INTERIX package from the master install program. If you need to install the SDK package at a later date, do the following:

1. Using File Manager, Explorer, or the **CMD.EXE** console window, go to the **\X86** directory on the Microsoft INTERIX CD-ROM.
2. Run **SETUP.EXE**.
3. Follow the instructions given by the setup program.
4. Select the components you want to install.
5. You will be asked if you want to install support for Microsoft Visual C/C++. Only answer “yes” if MSVC is already installed. If you answer “yes,” the setup program prompts for the name of the directory in which MSVC is installed; enter it into the field. (This enables the **cc** and **c89** compiler drivers to find MSVC.)

If you install Microsoft Visual C/C++ *after* installing the Microsoft INTERIX SDK, see the “Microsoft Compilers” section in the Installation chapter of the Microsoft INTERIX *Software Development Kit User’s Guide*.

Saving Disk Space

The Microsoft INTERIX products take up less disk if installed on an NTFS file system than if installed on a FAT file system. This is because the block size on NTFS is smaller than the FAT sector size.

You can save a few megabytes of disk space after the install by linking programs that use the same image. This will only work if you have installed Microsoft INTERIX on an NTFS file system. For example:

```
$ cd $INTERIX_ROOT/bin
$ rm ksh sh sh.exe
$ ln ksh.exe ksh
$ ln ksh.exe sh
$ ln ksh.exe sh.exe
```

The program files *csh*, **csh.exe**, and *tcsh* can be linked to the **tcsh.exe** program. The program *tar* can be linked to the *pax* program. The programs *pack*, *unpack*, *zcat*, *uncompress*, and *gunzip* can all be linked to the *gzip* program. The program *mail* can be linked to *mailx*.

You can also save space by not installing the PDF on-line documentation. The Adobe Acrobat Reader can read the files from the Microsoft INTERIX CD-ROM; they are stored in the directory **\PUBS**.

If you do not have access to an X11 display server, you might choose not to install the X11R5 clients.

Uninstalling Microsoft INTERIX

Uninstall programs are provided in case you should ever need to remove an INTERIX product from your machine. If you’re removing more than one INTERIX product, remove them in reverse order—start by removing the last product installed.

Before uninstalling, login as Administrator.

To uninstall Microsoft INTERIX products:

1. Reboot to ensure that the Microsoft INTERIX subsystem is not running. You may have to disable any Microsoft INTERIX services.
2. Run the appropriate uninstall program by opening the uninstall icon in the Microsoft INTERIX program group.
3. Restore the original POSIX subsystem files. These files are stored in the **%SYSTEMROOT%\SYSTEM32** directory. The three files are **POSIX.EXE**, **PSXSS.EXE**, and **PSXDLL.DLL**. You can either use the File Manager to rename the **.SAV** files back to **.EXE** extensions, or you can use the Microsoft INTERIX setup program.

MICROSOFT INTERIX PDF DOCUMENTATION

MICROSOFT INTERIX 2.2

SERVICES: THE TELNET DAEMON AND SYSLOGD

WORKING WITH TAPE DRIVES

X11 SERVER AND INTERIX X11 CLIENTS

MICROSOFT INTERIX SOFTWARE DEVELOPMENT KIT

Microsoft INTERIX Basics

This chapter describes the basics for getting started with Microsoft INTERIX. For more information about any of these products, see the appropriate *User's Guides*, which are provided on-line as PDF documents.

All examples assume you have installed Microsoft INTERIX in the default location, **C:\Interix** (for a new installation). Pathnames are written in the Microsoft INTERIX subsystem format: drives are prefixed with a double forward slash and a forward slash is used as the separator, so **C:\Interix** is referred to as **//C/Interix**.

The install directory for Microsoft INTERIX is also stored in the environment variable **OPENNT_ROOT**. In some documentation, you may see pathnames such as **\$OPENNT_ROOT/etc**; this refers to the directory **etc** in the INTERIX install directory.

Microsoft INTERIX PDF Documentation

All of the Microsoft INTERIX packages contain on-line documentation in Adobe's Portable Document Format (PDF). You must have installed a PDF reader to read these files. See the previous chapter for installation instructions.

The Microsoft INTERIX *Commands and Utilities User's Guide* contains some information on customizing the Acrobat display.

Note that in the PDF version of the User's Guides, the page numbers that appear after index entries and table of contents entries are links to the item being referenced. If you click on the page number, Acrobat will display that page.

Microsoft INTERIX 2.2 and the Microsoft INTERIX 2.2 Software Development Kit also provide a set of formatted reference pages that can be read with the Microsoft INTERIX *man* command.

Microsoft INTERIX 2.2

To start the Microsoft INTERIX KornShell shell, either run the Microsoft INTERIX *ksh* icon in the Microsoft INTERIX Program Group, or from a **CMD.EXE** DOS prompt, type:

```
C:\Interix\bin\ksh -l
```

(That's the letter ell, not the number one.) This gives you the POSIX-conformant KornShell.

When you start the KornShell with the **-l** option, it runs the commands in the *system-wide* startup file, **//C/Interix/etc/profile**, and then runs the commands in the file **.profile** in your home directory, if it exists.

To customize your personal Microsoft INTERIX environment (for example, to add command aliases or to set environment variables), create the file **.profile** in your home directory. You can use the file **//C/Interix/etc/profile.usr** as a template.

For more information about selecting a different login shell, setting your environment variables, modifying your startup files, or configuring your terminal display attributes, see the *INTERIX Commands and Utilities User's Guide*.

Services: The Telnet Daemon and Syslogd

Microsoft INTERIX 2.2 contains ports of the UNIX *telnetd* and *syslogd* programs.

Once you have installed *telnetd* as a service, remote users can log into your machine. The *telnetd* version allows multiple users to be logged in at once.

If you already have a telnet service installed (for example, the telnet service supplied with OPENNT 1.0 and 1.1), you must un-install it before installing *telnetd*.

To install the telnet daemon as a service:

1. Login as Administrator.
2. From a Microsoft INTERIX shell, install the service using the *service* command. The *passwd* is the password to the Administrator account; the command here is broken on two lines to fit the page.

```
$ service install -u Administrator -p passwd \  
-s auto //C/Interix/bin/telnetd
```

3. Once the service is installed, you must start it:

```
$ service start telnetd
```

To stop the service, login as Administrator and use the command

```
$ service stop telnetd
```

To remove the service, login as Administrator and use the command

```
$ service remove telnetd
```

The *syslogd* program records system messages from Microsoft INTERIX programs. (If *syslogd* is not running, programs write system messages to the file **\$INTERIX_ROOT/var/adm/log/logger**. This file, like all log files, can get very large. Check them and remove them occasionally.)

To install the system log daemon as a service:

1. Login as Administrator.
2. From a Microsoft INTERIX shell, install the service using the *service* command. The *passwd* is the password to the Administrator account; the command here is broken on two lines to fit the page.

```
$ service install -u Administrator -p passwd \  
//C/Interix/bin/syslogd -s auto
```

3. Once the service is installed, you must start it:

```
$ service start syslogd
```

Start and stop the service in the same way as for *telnetd*, but using the name “syslogd” instead of “telnetd”.

The services can also be stopped and started through the Windows NT Control Panel. Other Microsoft INTERIX programs can be run as services. See the Microsoft INTERIX *Commands and Utilities User's Guide* for more information.

Working With Tape Drives

Microsoft INTERIX provides access to tape drives on your Windows NT machine. The tape devices have the file names **/dev/tapex** and **/dev/ntapex**, where *x* is a digit from 0 to 9. The devices **/dev/tape0** and **/dev/ntape0** refer to the same tape drive, but using the **ntape** name forces the tape to be used in non-rewind mode. The utilities *tar*, *pax*, *dd*, and *mt* are often used when dealing with tape drives.

If you have trouble reading tapes created on UNIX systems, the problem may be the blocking factor or the block size. When writing tapes, BSD-derived systems have a different default behavior than System V-derived systems. See chapter 5 of the Microsoft INTERIX *Commands and Utilities User's Guide* and the *mt* reference page for more details.

The Microsoft INTERIX X11 Clients

This section describes some of the information you may need to use the X11 clients. With an X11 server installed, you can use a traditional X11 window manager on your Microsoft INTERIX systems instead of the Windows NT window manager. The *twm* window manager is provided with the other X11 clients.

Window Managers and Window Modes

To run an X11 window manager, you must configure your X11 server appropriately.

In general, X11 servers on Windows NT offer the choice of two *window modes*: single-window mode and multiple window mode.

In **single-window mode**, one application window is opened as the X root window. All X clients exist in this window. You must start a window manager to manage X clients in that window. X clients will not appear outside of this window, and no other configuration of the X11 server is needed.

In **multiple-window mode**, each X client is opened in its own window, controlled by the Microsoft Windows window manager (or an X Window window manager, if you start one). Each application window can be manipulated like any Windows application window. To run an X Window window manager in this mode, you will need to configure your X11 server to disable the local window manager (that is, Windows NT). To get the root menu options, you may also need to configure the system to pass mouse events directly to X11.

Even if you have disabled the local window manager, the Windows NT window manager will still handle all non-X11 programs. The configuration affects only X11 programs.

Microsoft INTERIX X11 Clients

Microsoft INTERIX includes a number of X11R5 client programs. Before you run any X client programs:

- You must be running an X11 server *before* you can run an X11 application. Typically, this means you start an X11 server from a “Startup” group or, after you login, from the X11 Server group.
- You must set the *DISPLAY* environment variable:

```
$ export DISPLAY=localhost:0
```

Or if you are logged in from a remote machine, set the *DISPLAY* environment variable to your remote machine name or IP address. For example, if the machine name is *unix_box*, type:

```
$ export DISPLAY=unix_box:0
```

To run an X client command such as *xcalc*, type the command on the Microsoft INTERIX console command line:

```
$ xcalc &
```

The directory **\$INTERIX_ROOT/usr/X11R5/bin** is included as part of the default *PATH* environment variable for the INTERIX shells, so all of the X client programs are directly executable.

Microsoft INTERIX X Clients on the Start Menu

The Win32 program **psxrun.exe** provides an interface between INTERIX subsystem binaries and the Win32 subsystem.

You may want to install on the start menu some X clients shipped with INTERIX. For example, to install *xcalc* as a shortcut:

1. Create a shortcut.
2. Open the Properties for the shortcut.

3. Add the following to the target in the Shortcut tab (line broken to fit):

```
C:\WINNT\system32\PSXRUN.EXE C:\Interix\bin\ksh -l -c
"/C/Interix/usr/X11R5/bin/xcalc -bg blue -fg yellow"
```

Microsoft INTERIX Software Development Kit

Because the tools provided in the Microsoft INTERIX Software Development Kit are UNIX tools, building an application is done in the same way as on a UNIX machine — with *gcc*, *cc*, *c89*, or *make*:

```
$ gcc files -o outfile -llibs
$ cc files -o outfile -llibs
$ c89 -o outfile objectfiles -llibs
$ make > make.out &2>1
```

Where *files* can be object files and/or source files. *Cc* and *c89* are provided as compiler drivers for Microsoft Visual C/C++, and will only work if MSVC is installed. Use *gcc* otherwise.

Because Microsoft INTERIX file systems make use of drive letters, you may need to change some macros in your makefile. For example:

```
BINDIR=/bin
INCDIR=/usr/include
LIBDIR=/usr/lib
```

becomes

```
BINDIR=//C/Interix/bin
INCDIR=//C/Interix/usr/include
LIBDIR=//C/Interix/usr/lib
```

or, more portably:

```
BINDIR=${INTERIX_ROOT}/bin
INCDIR=${INTERIX_ROOT}/usr/include
LIBDIR=${INTERIX_ROOT}/usr/lib
```

The INTERIX *make* program uses the same facility as BSD *make* to include other makefiles. This command includes the system makefile and the makefile “makefile.interix”:

```
.include <sys.mk>
.include "makefile.interix"
```

Building an X11 Application

To build X applications, you need to generate **Makefiles**; you can do this using either *xmkmf* or *imake*.

To create a **Makefile** using *xmkmf*, type the command *xmkmf* in the directory that contains your **Makefile**:

```
$ xmkmf
```

You will see the output:

```
mv Makefile Makefile.bak
imake -DUseInstalled
-I//C/INTERIX/usr/X11R5/lib/X11/config
```

You can invoke *imake* directly; use the options shown in the example:

```
$ imake -DUseInstalled  
-I//C/Interix/usr/X11R5/lib/X11/config
```

For a description of *imake*'s options, see the reference page for *imake*.

Once the makefile has been generated, build the utility:

```
$ make > make.out 2>&1
```

Both *imake* and *xmkmf* look for configuration and template files in the directory **\$INTERIX_ROOT/usr/X11R5/lib/X11/config**. You may need to edit the values in the file **OpenNT.cf** in that directory.

If you are using *cc* or *c89* (Microsoft Visual C/C++) to build applications, you need to change the line that reads “#define CcCmd gcc” to the appropriate command. See “Building X Applications” in the Microsoft INTERIX *Software Development Kit User’s Guide* for more information.

Note that the X libraries are static libraries, which means that X11 executables are big.

For more information about the utilities and libraries in the Microsoft INTERIX Software Development Kit and about the process of porting software to Microsoft INTERIX, see the Microsoft INTERIX *Software Development Kit User’s Guide* in the On-Line Library.