For Linux®

NetCOBOL

Installation Guide

FUJITSU

Fourth Edition: December 2008

For NetCOBOL for Linux Version 7.3

Preface

This installation guide includes information required before and during the installation and use of NetCOBOL for Linux.

NetCOBOL for Linux, which includes a compiler, runtime libraries, and a debugger, is intended for the development of business applications using the COBOL language.

Audience

This guide is intended for programmers who will install and use the NetCOBOL product.

How this Manual is Organized

This manual contains the following information:

- Media on which NetCOBOL is distributed
- Packages that make up the NetCOBOL product
- Input and output devices that are supported by NetCOBOL
- Disk and memory requirements for NetCOBOL
- How to install NetCOBOL, and related installation considerations
- Notes on the software and documentation

How to Use This Manual

For optimal use of this guide, readers should have a working knowledge of the Linux operating system and the COBOL programming language.

Conventions Used in this Manual

This manual uses the following typographic conventions.

Example of convention	Description
`Mkdir /cdrom`	What you type appears in Courier bold (contrasted with on-screen computer output).
`Cobol`	The names of commands, directories, files, and variables as well as on-screen computer output are depicted in Courier.
`Install_dir`	Courier italics indicate a command-line placeholder; replace with a real name or value.
...	Ellipses indicate the item immediately preceding can be specified repeatedly.
[`def`]	Indicates that the enclosed item may be omitted.
{`ABC`|`DEF`}	Indicates that one of the enclosed items delimited by | is to be selected.
The *sheet* acts as an application creation form.	Italics are occasionally used for emphasis.
"NetCOBOL User's Guide" See Table 2 "Required Disk Size".	References to other publications or sections within publications are in quotation marks.

Related Manuals

Related documentation is listed below:

- NetCOBOL Language Reference
- NetCOBOL for Linux User's Guide
- NetCOBOL for Linux Web Guide
- NetCOBOL for Linux File Access Routines User's Guide

Security

To ensure security when working in an environment that is connected to the Internet, it is important to correctly set up both the applications created with NetCOBOL and their operating environment.

To safeguard resources (such as databases, and input and output files), and definition and information files required for the operation of programs from illegal access and tampering, you need to restrict access to the resources by OS functions and programs. In particular, keep important resources in an intranet environment in which a firewall has been installed.

Although this product offers different communication functions (such as the simple communication interface facility, and the Web subroutines), only the Web subroutines have been designed for use with Internet services. Therefore, only use these other functions in environments that are not connected to the Internet, or in intranet environments in which firewalls have been installed and which have been constructed to prevent security breaches.

If you are using the Web subroutines with NetCOBOL on a Web server, use the Web server authorization apparatus and encryption communication function (SSL) to prevent illegal access or information being leaked or tampered with. Additionally, use the Web server access log to investigate and pursue any incidents of illegal access. For details, refer to the documentation for the Web server you are using.

You need to test applications created with NetCOBOL to ensure that even if malicious or careless data values are provided as input, no important data can be destroyed or sensitive information obtained.

High Risk Activity

The Customer acknowledges and agrees that the Product is designed, developed and manufactured as contemplated for general use, including without limitation, general office use, personal use, household use, and ordinary industrial use, but is not designed, developed and manufactured as contemplated for use accompanying fatal risks or dangers that, unless extremely high safety is secured, could lead directly to death, personal injury, severe physical damage or other loss (hereinafter "High Safety Required Use"), including without limitation, nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system. The Customer, shall not use the Product without securing the sufficient safety required for the High Safety Required Use. In addition, Fujitsu (or other affiliate's name) shall not be liable against the Customer and/or any third party for any claims or damages arising in connection with the High Safety Required Use of the Product.

Trademarks

Contents

Chapter 1. Distribution

NetCOBOL for Linux is distributed on CD-ROM.

Packages

The CD-ROM contains the following file:

```
/mnt/cdrom/LINUX
```

Note that `cdrom` is the name of the directory in which the file is stored.

A *package* is one or more programs that comprise a software function. Table 1.1 lists the product names and functions associated with NetCOBOL for Linux.

Table 1.1. NetCOBOL Packages

Package Name	Function
FJSVcbl	NetCOBOL Development Kit
FJSVcbr	NetCOBOL Runtime System

Prerequisite Software

The platforms currently formally supported by NetCOBOL for Linux are shown in Table 1.2.

Table 1.2. NetCOBOL for Linux Supported Platforms

Distribution	Locale
Red Hat Enterprise Linux (v.4 for x86)	C , en_US.UTF-8
Red Hat Enterprise Linux (v.4 for EM64T) 32-bit mode	C , en_US.UTF-8
Red Hat Enterprise Linux (v.5 for x86)	C , en_US.UTF-8
Red Hat Enterprise Linux (v.5 for Intel64) 32-bit mode	C , en_US.UTF-8

NetCOBOL products have been verified to run with the packages listed in the following tables. The tested release and the function for which the package is required are as shown.

Table 1.3. Red Hat Enterprise Linux AS (v.4 for x86)

Package	Tested Release	Required For
kernel	2.6.9-5	Basic NetCOBOL functions
coreutils	5.2.1-31	Basic NetCOBOL functions
bash	3.0-19.2	Basic NetCOBOL functions
tcsh	6.13-9	Basic NetCOBOL functions
glibc	2.3.4-2	Basic NetCOBOL functions
glibc-common	2.3.4-2	Basic NetCOBOL functions
rpm	4.3.3-7_nonptl	Basic NetCOBOL functions
binutils	2.15.92.0.2-10.EL4	Basic NetCOBOL functions
man	1.5o1-9	Reading man manuals
cups	1.1.22-0.rc1.9	Printing forms
gcc	3.4.3-9	Using the C language interface
glibc-devel	2.3.4-2	
glibc-headers	2.3.4-2	
xorg-x11-devel	6.8.2-1-23.EL	
make	3.80-5	Using the make command
grep	2.5.1-31	Using development tools
gawk	3.1.3-10.1	
httpd	2.0.52-9.ent	Using the www Server
modssl	2.0.52-9	
openssl	0.9.7a-43.1	

Table 1.4. Red Hat Enterprise Linux AS (v.4 for EM64T) Update 4

Package	Tested Release	Required For
kernel	2.6.9-42	Basic NetCOBOL functions
coreutils	5.2.1-31.4	Basic NetCOBOL functions
bash	3.0-19.3	Basic NetCOBOL functions
tcsh	6.13-9	Basic NetCOBOL functions
glibc	2.3.4-2.25	Basic NetCOBOL functions
glibc-common	2.3.4-2.25	Basic NetCOBOL functions
rpm	4.3.3-18_nonptl	Basic NetCOBOL functions
binutils	2.15.92.0.2-21	Basic NetCOBOL functions
man	1.5o1-9.rhel4	Reading man manuals
cups	1.1.22-0.rc1.9.11	Printing forms
gcc	3.4.6-3	Using the C language interface
glibc-devel	2.3.4-2.25	
glibc-headers	2.3.4-2.25	
xorg-x11-devel	6.8.2-1.EL.13.36	
make	3.80-6.EL4	Using the make command

grep	2.5.1-32.2	Using development tools
gawk	3.1.3-10.1	
httpd	2.0.52-25.ent	Using the www Server
modssl	2.0.52-9	
openssl	0.9.7a-43.10	

Table 1.5. Red Hat Enterprise Linux AS (v.5 for x86) (v.5 for Intel64)

Package	Tested Release	Required For
kernel	2.6.18-8.el5	Basic NetCOBOL functions
coreutils	5.97-12.1.el5	Basic NetCOBOL functions
bash	3.1-16.1	Basic NetCOBOL functions
tcsh	6.14-12.el5	Basic NetCOBOL functions
glibc	2.5-12	Basic NetCOBOL functions
glibc-common	2.5-12	Basic NetCOBOL functions
rpm	4.4.2-37.el5	Basic NetCOBOL functions
binutils	2.17.50.0.6-2.el5	Basic NetCOBOL functions
man	1.6d-1.1	Reading man manuals
cups	1.2.4-11.5.el5	Printing forms
gcc	4.1.1-52.el5	Using the C language interface
glibc-devel	2.5-12	
glibc-headers	2.5-12	
imake	1.0.2-3	
make	3.81-1.1	Using the make command
grep	2.5.1-54.2.el5	Using development tools
gawk	3.1.5-14.el5	
httpd	2.2.3-6.el5	Using the www Server
modssl	2.2.3-6.el5	
openssl	0.9.8b-8.3.el5	

Input and Output Devices

NetCOBOL for Linux supports several input and output devices. Support for storage devices, printers, displays, and keyboards are discussed in this section.

File Output Devices

Magnetic storage devices can be connected to Red Hat Enterprise Linux systems and used with COBOL file systems.

Chapter 2. Installing NetCOBOL for Linux

Only people who are registered as super users, on systems configured as servers, can install NetCOBOL for Linux and its related packages.

System Requirements

The disk and memory space required for installation and use of NetCOBOL are shown below.

Table 2.1. System Requirements

Item	Size
Required Disk Space	23 MB
Memory Required for Compilation	1.2 MB minimum
Memory Required for Execution, Runtime System	1.0 MB minimum

Package Installation

The installation of NetCOBOL for Linux uses the rpm command. Install it after confirming there is enough disk space. For necessary disk space for NetCOBOL for Linux, see Table 2.1 "System Requirements."

The standard installation directory is /opt. When the disk space is insufficient for the partition of /opt , it is possible to install it into a different directory structure. However, the symbolic link files are installed at /opt even when installing NetCOBOL for Linux into directories other than /opt.

1. Confirm whether the package has already been installed by the rpm command.

```
# rpm -q FJSVcbl FJSVcbr
```

Delete the package by the rpm command if the package has already been installed. For deleting the package, see "Deleting a package".

Note: If your system has a graphical RPM handler, you can use that instead.

2. Install the package using the rpm command.

```
# rpm -U /media/cdrom/LINUX/FJSVcbl-7.3.0-2.0.i486.rpm \
         /media/cdrom/LINUX/FJSVcbr-7.3.0-2.0.i486.rpm
```

Use –prefix if you wish to define a directory structure to precede the directory structure of the product.

Note: If your system has a graphical RPM handler, you can use that instead.

Setting Up the User Environment

NetCOBOL for Linux users have several options to select from when configuring their development environment. To specify what paths that libraries, manuals, messages, and other data are to be stored in, follow the examples below for setting the `PATH` and `MANPATH` environment variables.

The examples that are used in this section can be found in the following files:

```
 /opt/FJSVcbl/config/cobol.sh         (Bourne shell)
/opt/FJSVcbl/config/cobol.csh        (C shell)
```

For example, if you are using the C shell (csh), you must run:

```
csh
```

and then within the C shell you run:

```
source /opt/FJSVcbl/config/cobol.csh
```

to set up the environment.

If you are using the Bourne shell (`sh`), you would run `source` against the `cobol.sh` file instead.

The installer creates symbolic links of all files under `/opt/FJSVcbl`. The installation directory, therefore, need not be specified in the environment variables.

Deleting a Package

If you repartition the system, you will have to delete the NetCOBOL for Linux package and reinstall it in a new directory. To delete a package, for this or any other reason, issue the rpm command.

```
# rpm -e FJSVcbl FJSVcbr
```

Note: If your system has a graphical RPM handler, you can use that instead.

Chapter 3. Setting Up Your License

Use of NetCOBOL for Linux is managed by X-license software. To install this software you need about 2 MBytes of disk space. The installation directories are `/opt` and `/etc/opt`. You need the access rights of the Superuser.

Installing X-license

Install the software using the Linux commands:

```
# cd /
# tar -zxvf /media/cdrom/LINUX/xlicense.tar.gz
```

Applying for and Installing Your License

You need to request your full license by completing the registration form at: www.netcobol.com/registration

Within two business days you will receive your license document by email. It is an html document that you store unchanged in the directory:

```
/etc/opt/XSOlicense/licenses
```

It is important that you do not modify or reformat the .html license document that you receive. For example, if you were to view the license in a web browser, then view the source of the web page, then save the contents, your web browser would have reformatted the license, rendering it unusable.

Any modification to the license directory becomes effective within 30 seconds. Be careful to preserve your license file, as it is this document that enables your software. We suggest you keep a backup copy of this file and ensure that you preserve it if you ever uninstall the software.

Starting the License Server

The license server is started by the command:

```
# /opt/XSOlicense/bin/xlicense &
```

Removing the X-license Software

To remove the X-license software use the Linux command:

```
# rm -r /opt/XSOlicense /etc/opt/XSOlicense
```

Moving to Another Machine

If you wish to move NetCOBOL for Linux to another machine, you will need a new license file. Contact Fujitsu Computer Systems technical support at www.netcobol.com/support for instructions on performing this operation and to obtain a new license file.

License Validation and Troubleshooting

Once you have installed the above software and your unique license file, you can validate that the license software is functional by running xlicense with the -info option:

```
/opt/XSOlicense/bin/xlicense  -info
```

If the output contains any errors, additional information can often be found in the default log file generated by xlicense. The default log file is located here:

```
/etc/opt/XSOlicense/logfile.html
```

Common Problems and Solutions

- The default configuration of the xlicense software allows licenses to be stored on and provided from different servers in your local area network. Consequently, the xlicense software requires UDP ports 3074, 3075, 3174, and 3175 to be open. If this is not desirable or possible, you can change the default network browsing behavior of the xlicense server by modifying the /opt/XSOlicense/bin/xlicense file. The last line of the file looks like this by default:

  ```
  exec $XLICENSE_HOME/bin/_xlicense "$@"
  ```

  This can be changed to include a -localOnly flag as follows:

  ```
  exec $XLICENSE_HOME/bin/_xlicense -localOnly "$@"
  ```

- If you see file or directory access errors, make sure that the user account that started the xlicense process has read-permissions to both the license directory and the license file.
- If you receive error messages stating that certain libraries or other files cannot be found when attempting a compile, make sure that you have set up your environment variables as described in Chapter 2. If you echo $LD_LIBRARY_PATH and it is not defined or is other than /opt/FJSVcbl/lib/, then your environment variables may not be configured correctly. It is also possible to receive linker errors to shared object files that do not exist in

your environment if you happen to be using an unsupported operating system.

- If you get an error indicating that your license is not for this host, confirm that the MAC address contained in your license file matches that of the network adapters defined on your system. To view this information for your system, enter:

  ```
  /sbin/ifconfig -a | grep HWaddr
  ```

  You can visually compare this with your license's MAC address by entering:

  ```
  more /etc/opt/XSOlicense/licenses/*.html | grep Ethernet
  ```

  Note that MAC addresses are not case sensitive.

- Special note for users with bonded network adapters: If you have bonded adapters defined on your system, the xlicense service may be unable to validate your license. Contact NetCOBOL support for a software update to fix this problem.

Chapter 4. Software Notes

A) Not supported in this product:

- Presentation file function
- Screen function
- WITH phrase on PROCEDURE DIVISION, CALL statement, and ENTRY statement.
- OLE, COM, COB Class
- Internal floating-point item compatibility with other languages
- Stack size 2MB over program (see "*Cautions*" in Chapter 4 of the User's Guide for stack usage calculations)
- SORT and MERGE function to High Capacity Files

B) COPY statements

- A compile unit cannot end in a COPY statement. If an END PROGRAM statement is present in a COPY file at the end of the compile unit, compile errors cannot be interpreted correctly and the COBOL Debugger cannot be relied upon.

C) Warning message on RAISING phrase

Compile error "JMN3448I-W" may be issued on a RAISING phrase in an EXIT METHOD statement. This is a result of adding the RAISING phrase to the PROCEDURE DIVISION header. However, the error does not affect program execution. Refer to the "Procedure Division Header" section in the "Fujitsu COBOL Language Reference" for details.

D) Program, class and method names

- The COBOL system uses, internally, names that start with two underscores (__). A program, class or method name that starts with two successive underscores may already have been used as an internal name.
- The following names cannot be used: MAIN, main, JMP???, COB_???, COBOL_???, COBW3_???

E) Division by zero

If a program divides by zero, an application error occurs.

F) Overflow of floating-point items

If the value of a single or double precision floating-point item overflows, unpredictable errors may occur.

G) Prototype method

A shared object cannot be made by the prototype method.

You must link prototype methods as the same shared object as the definition class.

H) Filename on COBOL file system

The following characters cannot be used in a file name:

* , ?

I) Using I/O in multithread mode

If two threads try to open the same file at the same time, you can enter a deadlock state on an OPEN statement or any statement that accesses the same resource.

J) Using the File Utility

The COBOL File Utility is not offered with this release.

K) Output standard ERROR to a file

In the multithreaded environment, when a standard output of the system and the standard error output are written to the same file by specifying redirection, the content of the output file is not guaranteed.

Specify the file output to the CBR_MESSOUTFILE environment variable when you output the result of a DISPLAY verb associated with the run time system outputs and function name SYSERR to a file.

L) Using the debugger

Specify NODLOAD for translating when you use the coverage function of the debugger when debugging a program using an object-oriented function.

M) Using COBOL commands

The number of parameters that can be registered is recommended to be 500 or less.

N) Using FUNCTION CURRENT-DATE

FUNCTION CURRENT-DATE needs the TZ environment variable. Execute

```
man tzset
```

if you need more information of the TZ environment variable.

Example [sh]:

```
TZ="CST –6"; export TZ
```

Example [csh] :

```
setenv TZ "CST –6"
```

O) Using the ODBC interface to access remote databases

NetCOBOL for Linux supports ODBC for database access. But there are potential limitations and problems that might occur for SQL functions offered by NetCOBOL for Linux. This can be caused by conflicting functionality between the ODBC driver and the database.

Thus, NetCOBOL for Linux cannot guarantee the results of SQL operations. There may be conflicts between the ODBC driver, the database, and the COBOL application. Your application must be thoroughly tested in conjunction with the ODBC driver and the database.

P) Using Intel64(EM64T) Operating System

NetCOBOL for Linux is supported in 32 bit compatible mode only.

- When the gcc command is used, it is necessary to specify the following options.
  -m32 -mtune=i386
- When the ld command is used, it is necessary to specify the following options.
  -m elf_i386 -b elf32-i386 --oformat elf32-i386 -L/usr/lib

The ld command and the Dynamic Loader do not support combinations of 32 bit and 64 bit objects.