BEA eLink for Mainframe TCP

Installation Guide

BEA eLink for Mainframe TCP Version 3.1
Document Edition 3.1
August 1999

Copyright

Restricted Rights Legend

Trademarks or Service Marks

BEA® eLink™ for Mainframe TCP Installation Guide

Document Edition	Part Number	Date	Software Version
3.1	825-001007-002	August 1999	BEA eLink for Mainframe TCP 3.1
1.0	825-001007-001	April 1999	BEA eLink for Mainframe TCP 3.0

Contents

Preface

1. Installing BEA eLink TCP for TUXEDO

2. Installing BEA eLink TCP for IMS

3. Installing BEA eLink TCP for CICS

A. Changing Resource Names

Preface

BEA eLink for Mainframe TCP (hereafter referenced as eLink TCP) is a gateway connectivity feature that enables application programs on BEA TUXEDO systems to perform various non-transactional tasks with application programs that reside on different kinds of computers.

Purpose of This Document

This document explains how to install all components of the eLink TCP product.

- Installing BEA eLink TCP for TUXEDO
- Installing BEA eLink TCP for IMS
- Installing BEA eLink TCP for CICS

Who Should Read This Document

This document is primarily for system administrators who install and configure the eLink TCP product.

Programmers and system administrators who work with eLink TCP should be familiar with the concept of distributed multi-tier client/server processing.

In addition, all readers should be familiar with the following:

- The operating system
- BEA TUXEDO software

Finally, system administrators should be familiar with TCP/IP networking.

How This Document Is Organized

The *BEA eLink TCP Installation Guide* is organized as follows:

- Chapter 1, “Installing BEA eLink TCP for TUXEDO,” provides information for installing the eLink TCP for TUXEDO component on a UNIX, OS/390, or Windows NT system.
- Chapter 2, “Installing BEA eLink TCP for IMS,” provides information for installing the eLink TCP for IMS component.
- Chapter 3, “Installing BEA eLink TCP for CICS,” provides information for installing the eLink TCP for CICS component.

Document Conventions

The following documentation conventions are used throughout this manual.

Item	Examples
Variable names	Variable names represent information you must supply or output information that can change; they are intended to be replaced by actual names. Variable names are displayed in italics and can include hyphens but not underscores. The following are examples of variable names in text. *error-file-name* The *when-return* value...
Function names in text	C function names are displayed in lower case type and can include parentheses and possibly underscores, as follows. routine_name() COBOL function or subprogram names are displayed in uppercase type without underscores or hyphens, as follows. ROUTINENAME()

Item	Examples
Symbolic constants for languages (keywords, error codes, and flags)	C symbolic constants are displayed in uppercase type and can include underscores, as follows. CONSTANT_NAME COBOL symbolic constants are displayed in uppercase type and can include hyphens, as follows. CONSTANT-NAME
User input and screen output	For screen displays and other examples of input and output, user input appears as in the first of the following lines; system output appears as in the second through fourth lines. **`dir c:\accounting\data`** Volume in drive C is WIN_NT_1 Volume Serial Number is 1234-5678 Directory of C:\ACCOUNTING\DATA
Syntax	Code samples can include the following elements: ♦ Variable names can include hyphens but not underscores (e.g., *error-file-name)* ♦ Optional items are enclosed in square brackets: [ ]. If you include an optional item, do not code the square brackets. ♦ A required element for which alternatives exist is enclosed in braces {}. The alternatives are separated by the pipe (vertical bar) character: |. You must include only one of the alternatives for that element. Do not code the braces or pipe character. ♦ An ellipsis (...) indicates that the preceding element can be repeated as necessary. C synopsis: int tpacall(char *svc, char *data, long len, long flags) COBOL statement: CALL “TPACALL” USING TPSVCDEF-REC TPTYPE-REC DATA-REC TPSTATUS-REC.
Omitted code	An ellipsis (...) is used in examples to indicate that code that is not pertinent to the discussion is omitted. The ellipsis can be horizontal or vertical.

Related Documentation

The following sections list the documentation provided with the eLink TCP for IMS software, and other publications related to online transaction processing (OLTP) technology.

BEA eLink TCP Documentation

The eLink TCP documentation consists of the following items:

- *BEA eLink TCP for TUXEDO User Guide*
- *BEA eLink TCP for CICS User Guide*
- *BEA eLink TCP for IMS User Guide*
- *BEA eLink TCP Release Notes*

Product Manuals

The following BEA publications are also available:

- *TUXEDO System 6 Reference Manual*
- *TUXEDO System 6 Programmer's Guide, Volumes 1 and 2*

Other Publications

For more information about OLTP technology, refer to the following books:

- *The TUXEDO System* (Andrade, Carges, Dwyer, Felts)
- *TUXEDO: An Open Approach to OLTP* (Primatesta)
- *Building Client/Server Applications Using TUXEDO* (Hall)

Contact Information

The following sections provide information about how to obtain support for the documentation and software.

Documentation Support

If you have questions or comments on the documentation, you can contact the BEA Information Engineering Group by e-mail at **docsupport@beasys.com**. (For information about how to contact Customer Support, refer to the following section.)

Customer Support

If you have any questions about this version of eLink TCP, or if you have problems installing and running eLink TCP, contact BEA Customer Support through BEA WebSupport at `www.beasys.com`. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

1 Installing BEA eLink TCP for TUXEDO

The installation procedure for BEA eLink for Mainframe TCP for TUXEDO (hereafter referenced as eLink TCP for TUXEDO) is slightly different for each platform on which the product can be installed. Please follow the directions for the appropriate platform. The following information assists you in successfully installing this product:

- Pre-Installation Considerations
- Installing eLink TCP for TUXEDO
  - Installing on Unix-based Platforms
  - Installing on OS/390 UNIX
  - Installing on a Windows NT Platform
- Distribution Libraries and Executables
- Uninstalling eLink TCP for TUXEDO on Windows NT

Pre-Installation Considerations

The eLink TCP for TUXEDO product runs on UNIX-based and Windows NT platforms. Complete the following tasks prior to installing the eLink TCP for TUXEDO software:

- Read the *BEA eLink TCP Release Notes*

- Install and verify the operation of one of the following:
  - BEA TUXEDO 6.4 or 6.5
  - BEA WebLogic Enterprise 4.0 (formerly known as BEA M3 2.2)
- Install and verify the TCP/IP stack software

Installing eLink TCP for TUXEDO

The eLink TCP for TUXEDO product runs on Unix-based platforms, OS/390 UNIX, and Windows NT. Refer to the appropriate platform sections that follow for installation instructions:

- Installing on Unix-based Platforms
- Installing on OS/390 UNIX
- Installing on a Windows NT Platform

Installing on Unix-based Platforms

The following steps install eLink TCP for TUXEDO on a Unix-based platform.

Note: For installation instructions for OS/390, refer to the "Installing on OS/390 UNIX" section.

1. Log on as root to install eLink TCP for TUXEDO.

Listing 1-1 Log On as Root

```
$ su -
Password:
```

2. Mount the CD-ROM and change directories to the top-level directory on the CD-ROM. Be sure to install eLink TCP for TUXEDO in the TUXEDO directory. The following example mounts the CD-ROM drive on a UNIX-based system.

Listing 1-2 Mount the CD-ROM Drive

```
# ls -l /dev/cdrom
total 0
brw-rw-rw-    1 root     sys       22,  0 Sep 16 10:55 c1b0t0l0
# mount -r -F cdfs /dev/cdrom/c1b0t0l0 /mnt
# cd /mnt
# ls
install.sh          winnt
```

3. Run the installation script by typing the following command.

Listing 1-3 Run the Installation Script

```
# sh ./install.sh
```

4. The installation script prompts you for responses. Listing 1-4 is a sample installation for the HP-UX platform.

Listing 1-4 Sample HP-UX Installation

```
YourName@YourMachine-> install.sh

01) alpha/dux40      02) hp/hpux1020      03) hp/hpux11
04) ibm/aix43        05) seq/dynix442     06) sgi/irix65
07) sun5x/sol26      08) sun5x/sol7

Install which platform's files? [01-8, q to quit, l for list]: 2

** You have chosen to install from hp/hpux1020 **

BEA eLink for Mainframe TCP Release 3.1
```

```
This directory contains the BEA eLink for Mainframe TCP System for
HP-UX 10.20 on 9000/800 series.


Is this correct? [y,n,q]: y

To terminate the installation at any time
press the interrupt key,
typically <del>, <break>, or <ctrl+c>.

The following packages are available:

  1    tcp           BEA eLink for Mainframe TCP

Select the package(s) you wish to install (or 'all' to install
all packages) (default: all) [?,??,q]: 1

BEA eLink for Mainframe TCP
(9000) Release 3.1
Copyright (c) 1999 BEA Systems, Inc.
All Rights Reserved.
Distributed under license by BEA Systems, Inc.
BEA eLink is a trademark of BEA Systems, Inc.


Directory where TCP files are to be installed
(Enter your WLE or TUXEDO directory path) [?,q]: /bea/work/TUX65

Using /bea/work/TUX65 as the TCP base directory

Determining if sufficient space is available ...
1050 blocks are required
8801556 blocks are available to /bea/work /TUX65

Unloading /cmhome/dist/tundra-2/hp/hpux1020/tcp/TCPT65.Z ...
bin/GWIDOMAIN
bin/lic.sh
lib/ConvMVS.sl
lib/ConvMVSC.sl
lib/WrapTPS.sl
lib/WrapTPSD.sl
lib/libctxcp_10.sl
locale/C/LIBGWI.text
locale/C/LIBGWI_CAT
udataobj/codepage/00819x00037
udataobj/codepage/00819x00273
udataobj/codepage/00819x00278
udataobj/codepage/00819x00284
udataobj/codepage/00819x00285
```

```
udataobj/codepage/00819x00297
udataobj/codepage/00819x00500
udataobj/codepage/00819x00860
udataobj/codepage/00819x01047
udataobj/codepage/00912x00870
udataobj/codepage/none
udataobj/codepage/tuxedo
1030 blocks
... finished

Changing file permissions...
... finished

If your license file is accessible, you may install it now.
Install license file? [y/n]: n

Please don't forget to use lic.sh located in your product bin
directory to install the license file from the enclosed floppy.
Refer to your product Installation Guide for details on how to do
this.


Installation of BEA eLink for Mainframe TCP was successful.

Please don't forget to fill out and send in your registration card.
```

5. Unmount the CD-ROM by typing a command similar to the following.

Listing 1-5 Unmount the CD-ROM

```
# cd /
# umount /mnt
# exit
$ exit
```

6. Use a text editor to verify that the following line in the `udataobj/DMTYPE` file is correct. If it is not, then delete it and replace it with the following new text.

Listing 1-6 Verify IDOMAIN Entry in udataobj/DMTYPE File

```
IDOMAIN:-lgwi:-ldl -lnwi -lnws -lnwi::
```

7. Use a text editor to verify that the following line in the `locale/CATNAMES` file is correct. If it is not, then delete it and replace it with the following new text.

   Note: This step does not apply for BEA WebLogic Enterprise 4.2.

Listing 1-7 Verify LIBGWI_CAT Entry in locale/CATNAMES File

```
41      LIBGWI_CAT
```

Installing on OS/390 UNIX

The following steps install eLink TCP for TUXEDO on a OS/390.

1. Load the CD on a machine that has FTP access to the OS/390 host.
2. FTP the file `uss390.tar` (in binary mode) from the `ibmoe` directory on the CD to the working directory on the OS/390 host.
3. Unmount the CD-ROM by typing a command similar to the following.

Listing 1-8 Unmount the CD-ROM

```
# cd /
# umount /mnt
# exit
$ exit
```

4. To extract the installation script from the tar file, use the following command.

Listing 1-9 tar Command to Extract the install.sh Script

```
tar xvf uss390.tar
```

Executing the tar command extracts from the tar file the `install.sh` script and subdirectories containing the software to be installed.

5. Execute the install.sh script using the following command.

Listing 1-10 Command to Execute the Installation Script

```
sh install.sh
```

6. The installation script prompts you for responses. Listing 1-4 is a sample installation for the SUN Solaris platform.

Listing 1-11 Sample OS/390 Installation

```
01) uss390/uss390r5

Install which platform's files? [01-     1 , q to quit, l for list]:
1

** You have chosen to install from uss390/uss390r5 **

BEA eLink for Mainframe TCP Release 3.1

This directory contains the BEA eLink for Mainframe TCP System for
IBM OS/390 Unix System Services R5 on IBM S/390.


Is this correct? [y,n,q]: y

To terminate the installation at any time
press the interrupt key,
typically <del>, <break>, or <ctrl+c>.

The following packages are available:
```

```
  1    tcp           BEA eLink for Mainframe TCP

Select the package(s) you wish to install (or 'all' to install
all packages) (default: all) [?,??,q]: 1

BEA eLink for Mainframe TCP
(USS390) Release 3.1
Copyright (c) 1999 BEA Systems, Inc.
All Rights Reserved.
Distributed under license by BEA Systems, Inc.
BEA eLink is a trademark of BEA Systems, Inc.


Directory where TCP files are to be installed
(Enter your WLE or TUXEDO directory path) [?,q]: /bea/work/tux65

Using /bea/work/tux65 as the TCP base directory

Determining if sufficient space is available ...
35424 blocks are required
1686552 blocks are available to /bea/work/tux65

Unloading /bea/work/inst/uss390/uss390r5/tcp/TCPT65.Z ...
3593 blocks
x bin/GWIDOMAIN
x bin/lic.sh
x lib/ConvMVS.so
x lib/ConvMVSC.so
x lib/WrapTPS.so
x lib/WrapTPSD.so
x lib/libctxcp_10.dll
x locale/C/LIBGWI.text
x locale/C/LIBGWI_CAT
x udataobj/codepage/00819x00037
x udataobj/codepage/00819x00273
x udataobj/codepage/00819x00278
x udataobj/codepage/00819x00284
x udataobj/codepage/00819x00285
x udataobj/codepage/00819x00297
x udataobj/codepage/00819x00500
x udataobj/codepage/00819x00860
x udataobj/codepage/00819x01047
x udataobj/codepage/00912x00870
x udataobj/codepage/none
x udataobj/codepage/tuxedo
... finished

Changing file permissions...
```

```
... finished

If your license file is accessible, you may install it now.
Install license file? [y/n]:
Please don't forget to use lic.sh located in your product bin
directory to install the license file from the enclosed floppy.
Refer to your product Installation Guide for details on how to do
this.


Installation of BEA eLink for Mainframe TCP was successful.

Please don't forget to fill out and send in your registration card.
```

7. Use a text editor to verify that the following line in the `udataobj/DMTYPE` file is correct. If it is not, then delete it and replace it with the following new text.

Listing 1-12 Verify IDOMAIN Entry in udataobj/DMTYPE File

```
IDOMAIN:-lgwi:-ldl -lnwi -lnws -lnwi::
```

8. Use a text editor to verify that the following line in the `locale/CATNAMES` file is correct. If it is not, then delete it and replace it with the following new text.

Listing 1-13 Verify LIBGWI_CAT Entry in locale/CATNAMES File

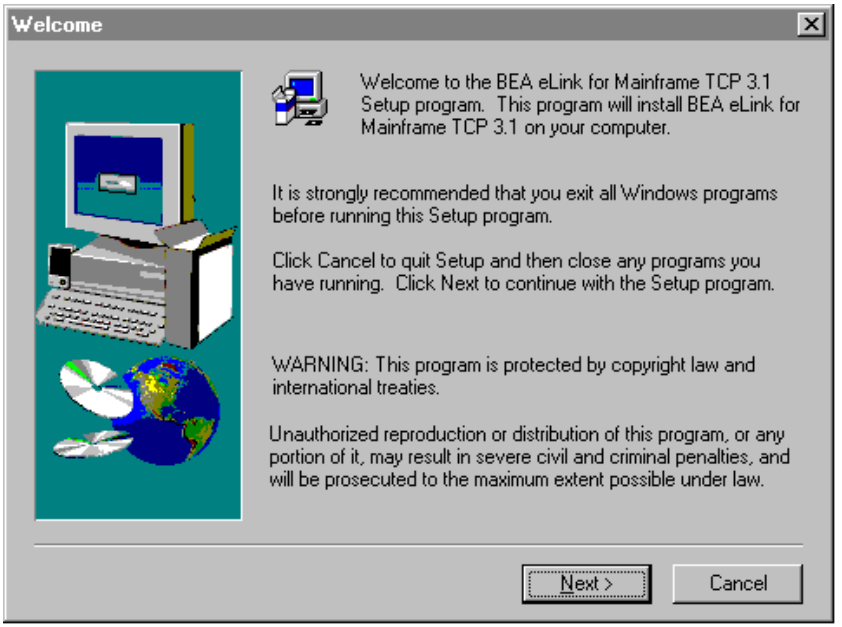
```
41     LIBGWI_CAT
```

Installing on a Windows NT Platform

The following steps install eLink TCP for TUXEDO on a Windows NT system.

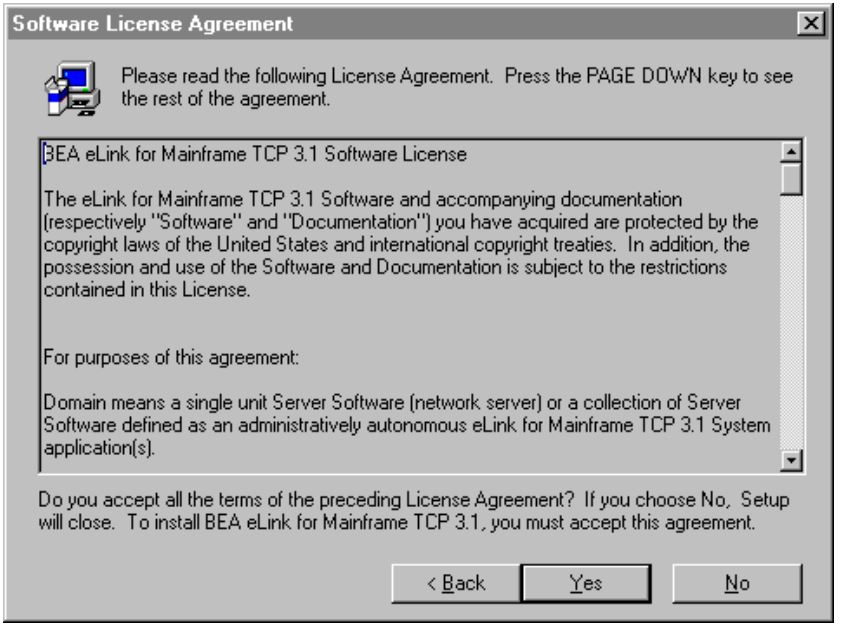
1. Insert the product CD-ROM and click the **Run** option from the **Start menu**. The **Run** window displays. Click the **Browse** button to select the CD-ROM drive. Select the `winnt` directory and select the `Setup.exe` program. Click **OK** to run the executable and begin the installation. The following **Welcome** screen displays. Click **Next** to continue with the installation.

Figure 1-1 Welcome Screen



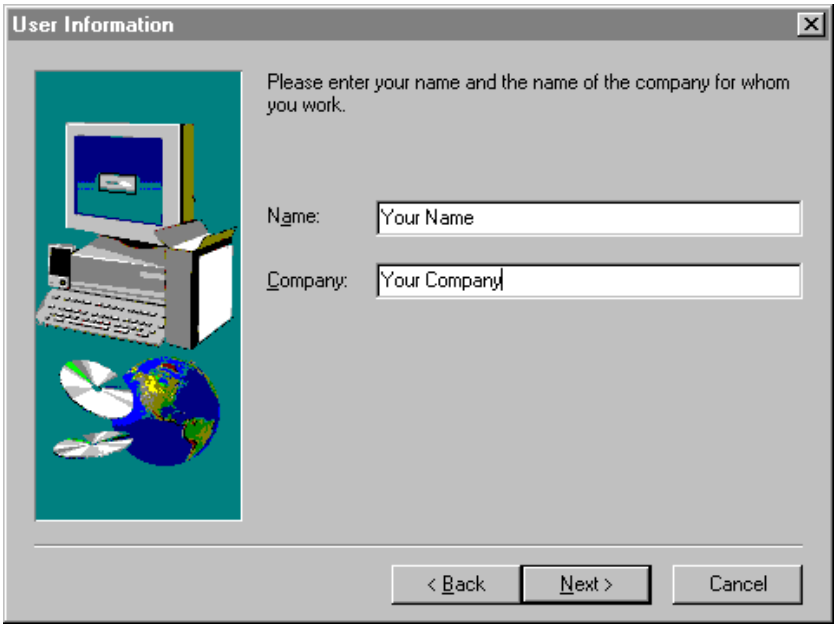
2. The BEA Software License Agreement displays. Click **Yes** to accept the terms of the agreement and continue with the product installation. Click **No** to exit the installation process.

Figure 1-2 BEA Software License Agreement

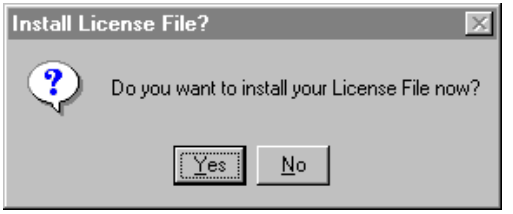


3. The **User Information** screen displays after the License Agreement. Enter the name of the system administrator in the **Name** field. Enter the name of your company in the **Company** field. Click **Next** to continue with the installation.

Figure 1-3 User Information



4. The **Install License File?** screen displays next.

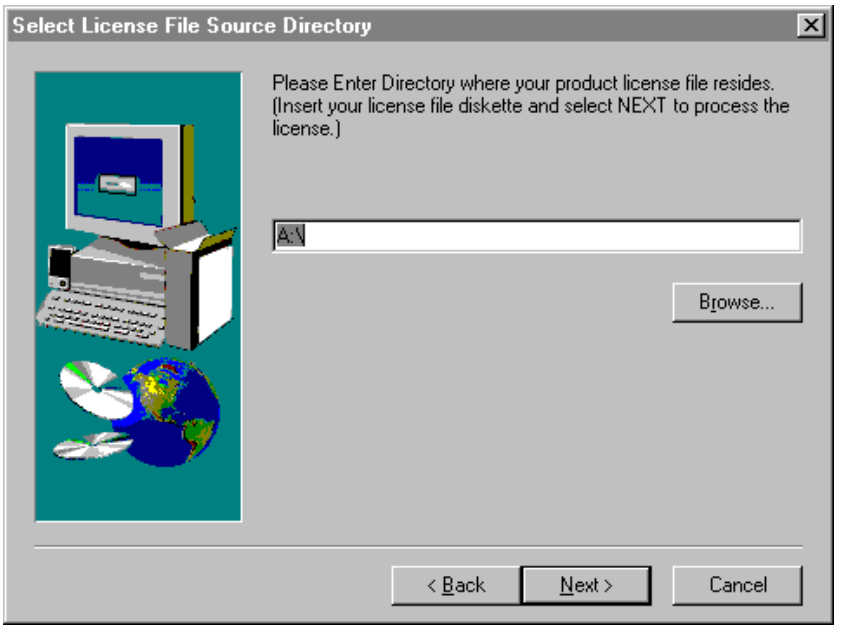


   a. Click **Yes** to install the license file now. The **Select License File Source Directory** screen displays. Proceed to step 5.

   b. Click **No** to continue the installation without installing the license file. If you click **No**, you must append the license file prior to using the product. Proceed to step 6.

5. The **Select License File Source Directory** screen displays. Enter the directory path where your license file resides. Typically, the license file is on a floppy disk. You can browse and select directories by clicking the **Browse** button. Click **Next** to continue with the installation.

Figure 1-4 Select License File Source Directory



If the directory you specify does not contain the license file, the following message displays.

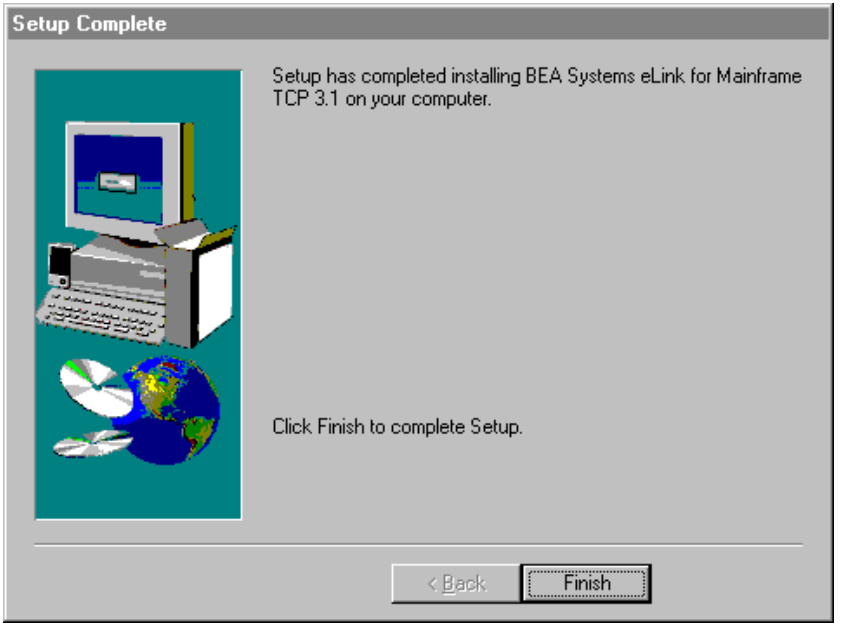
a. Click **Yes** to return to the **Select License File Source Directory** screen and specify another directory.

b. Click **No** to continue the installation without installing the license file. If you click **No**, you must append the license file prior to using the product.

6. A progress bar displays showing the status of the installation. You may abort the installation process anytime prior to completion by clicking the **Cancel** button.

7. The **Setup Complete** screen displays notifying you that the eLink TCP product is installed on your system. Click **Finish** to complete the setup process.

Figure 1-5 Setup Complete



8. Use a text editor, such as Microsoft NotePad, to verify that the following line in the `udataobj/DMTYPE` file is correct. If it is not, then delete it and replace it with the following new text.

Listing 1-14 Verify IDOMAIN Entry in udataobj/DMTYPE File

```
IDOMAIN;;;;
```

9. Use a text editor to verify that the following line in the `locale/CATNAMES` file is correct. If it is not, then delete it and replace it with the following new text.

   Note: This step does not apply for BEA WebLogic Enterprise 4.2.

10. Use a text editor to verify that the following line in the `locale/CATNAMES` file is correct. If it is not, then delete it and replace it with the new text as shown in Listing 1-15.

Listing 1-15 Verify LIBGWI_CAT Entry in locale/CATNAMES File

```
41      LIBGWI_CAT
```

Distribution Libraries and Executables

The eLink TCP CD-ROM contains the following libraries and executable programs. After installing the eLink TCP software, verify that these libraries and programs are installed on your system.

HP-UX 10.20 or 11.00

Verify that the following files are installed by eLink TCP for TUXEDO.

Directory	Files
/bin	GWIDOMAIN lic.sh
/lib	libctxcp_10.sl ConvMVS.sl ConvMVSC.sl WrapTPS.sl WrapTPSD.sl
/locale/C	LIBGWI_CAT LIBGWI.text

Directory	Files
/udataobj/codepage	00819x00037 00819x00278 00819x00285 00819x00500 00819x01047 00819x00273 00819x00284 00819x00297 00819x00860 00912x00870 none tuxedo

SUN Solaris 2.6 or 7

Verify that the following files are installed by eLink TCP for TUXEDO.

Directory	Files
/bin	GWIDOMAIN lic.sh
/lib	libctxcp_10.so ConvMVS.so.60 ConvMVSC.so.60 WrapTPS.so.60 WrapTPSD.so.60
/locale/C	LIBGWI_CAT LIBGWI.text

Directory	Files
/udataobj/codepage	00819x00037 00819x00278 00819x00285 00819x00500 00819x01047 00819x00273 00819x00284 00819x00297 00819x00860 00912x00870 none tuxedo

IBM AIX 4.3

Verify that the following files are installed by eLink TCP for TUXEDO.

Directory	Files
/bin	GWIDOMAIN lic.sh
/lib	libctxcp_10.so ConvMVS.so.60 ConvMVSC.so.60 WrapTPS.so.60 WrapTPSD.so.60
/locale/C	LIBGWI_CAT LIBGWI.text

Directory	Files
/udataobj/codepage	00819x00037 00819x00278 00819x00285 00819x00500 00819x01047 00819x00273 00819x00284 00819x00297 00819x00860 00912x00870 none tuxedo

DEC Alpha UNIX 4.0

Verify that the following files are installed by eLink TCP for TUXEDO.

Directory	Files
/bin	GWIDOMAIN lic.sh
/lib	libctxcp_10.so ConvMVS.so ConvMVSC.so WrapTPS.so WrapTPSD.so
/locale/C	LIBGWI_CAT LIBGWI.text

Directory	Files
/udataobj/codepage	00819x00037 00819x00278 00819x00285 00819x00500 00819x01047 00819x00273 00819x00284 00819x00297 00819x00860 00912x00870 none tuxedo

SEQUENT DYNIX 4.4.2

Verify that the following files are installed by eLink TCP for TUXEDO.

Directory	Files
/bin	GWIDOMAIN lic.sh
/lib	libctxcp_10.so ConvMVS.so.60 ConvMVSC.so.60 WrapTPS.so.60 WrapTPSD.so.60
/locale/C	LIBGWI_CAT LIBGWI.text

Directory	Files
/udataobj/codepage	00819x00037 00819x00278 00819x00285 00819x00500 00819x01047 00819x00273 00819x00284 00819x00297 00819x00860 00912x00870 none tuxedo

SGI IRIX 6.5

Verify that the following files are installed by eLink TCP for TUXEDO.

Directory	Files
/bin	GWIDOMAIN lic.sh
/lib	libctxcp_10.so.60 ConvMVS.so.60 ConvMVSC.so.60 WrapTPS.so.60 WrapTPSD.so.60
/locale/C	LIBGWI_CAT LIBGWI.text

Directory	Files
/udataobj/codepage	00819x00037 00819x00278 00819x00285 00819x00500 00819x01047 00819x00273 00819x00284 00819x00297 00819x00860 00912x00870 none tuxedo

IBM OS/390 UNIX

Verify that the following files are installed by eLink TCP for TUXEDO.

Directory	Files
/bin	GWIDOMAIN lic.sh
/lib	libctxcp_10.dll ConvMVS.so ConvMVSC.so WrapTPS.so WrapTPSD.so
/locale/C	LIBGWI_CAT LIBGWI.text

Directory	Files
/udataobj/codepage	00819x00037 00819x00278 00819x00285 00819x00500 00819x01047 00819x00273 00819x00284 00819x00297 00819x00860 00912x00870 none tuxedo

Windows NT 4.0 (Intel and Alpha)

Verify that the following files are installed by eLink TCP for TUXEDO.

Directory	Files
/bin	GWIDOMAIN.exe ctxcp_10.dll libgwi.dll ConvMVS.dll ConvMVSC.dll WrapTPS.dll WrapTPSD.dll
/locale/C	LIBGWI_CAT LIBGWI.text

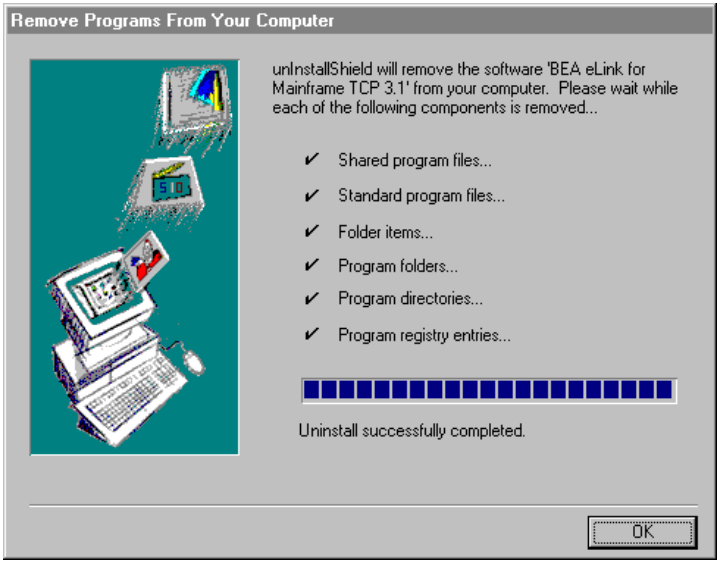
Directory	Files
/udataobj/codepage	00819x00037 00819x00278 00819x00285 00819x00500 00819x01047 00819x00273 00819x00284 00819x00297 00819x00860 00912x00870 none tuxedo

Uninstalling eLink TCP for TUXEDO on Windows NT

The following steps uninstall the eLink TCP for TUXEDO product on a Windows NT system.

1. Access the **Control Panel** window from the **Start>Settings>Control Panel** menu option.
2. Double click the **Add/Remove Programs** option from the Control Panel listings to access the **Add/Remove Programs Properties** window.
3. In the **Add/Remove Programs Properties** window, select **BEA eLink TCP 3.1** from the program list and click the **Add/Remove** button.
4. The uninstall process for this product begins. The **Remove Programs From Your Computer** screen displays. Click **OK** to complete the uninstall process.

Figure 1-6 Removing Programs From Your Computer



2 Installing BEA eLink TCP for IMS

Installing BEA eLink for Mainframe TCP for IMS (hereafter referenced as eLink TCP for IMS) consists of the following basic tasks.

1. Pre-Installation Considerations
2. Allocate OS/390 Data Sets
3. Unload the Distribution Files
4. Link-Edit eLink TCP for IMS
5. Define eLink TCP for IMS to IMS
6. Create an eLink TCP for IMS Configuration File
7. Create JCL to Run eLink TCP for IMS
8. Start BEA eLink TCP for IMS
9. Test the Installation

Pre-Installation Considerations

Complete the following tasks prior to installing the eLink TCP for IMS software:

- Read the *BEA eLink TCP Release Notes*
- Install and verify the TCP/IP stack software

Allocate OS/390 Data Sets

The directory structures on the eLink TCP for IMS software CD include the following partitioned datasets.

Partitioned Datasets (PDS)	Format	Contents
CONTROL	ASCII	Sample IMS definitions, sample BEA eLink TCP for IMS configuration files
INCLUDE	ASCII	C header files, COBOL copybooks for use by IMS Client and Server transactions
JCL	ASCII	Job Control Language for linking and executing BEA eLink TCP for IMS
SOURCE	ASCII	Sample IMS Client and Server transactions for use with BEA eLink TCP for IMS
OBJECT	BINARY	Object component files of BEA eLink TCP for IMS
MSGCAT (sequential dataset)	BINARY	File containing runtime messages that eLink TCP for IMS issue

FTP the JCL file INSTALL. Execute this job to allocate datasets using high-level qualifiers appropriate to your installation. Allocate the following datasets to receive the eLink TCP for IMS distribution files (*hlq1* and *hlq2* are dataset name high-level qualifiers appropriate to your installation).

Listing 2-1 Allocating OS/390 Data Sets

```
DSNAME: hlq1.hlq2.CONTROL
DSORG: PO
DIRBLKS: 10
RECFM: FB
LRECL: 80
BLKSIZE: 23440 (or other valid blocksize)
SPACE: 2 tracks
Usage: IMS sample definitions, sample configuration file

DSNAME: hlq1.hlq2.INCLUDE
DSORG: PO
DIRBLKS: 10
RECFM: FB
LRECL: 80
BLKSIZE: 23440 (or other valid blocksize)
SPACE: 2 tracks
Usage: C header file, COBOL copybooks

DSNAME: hlq1.hlq2.JCL
DSORG: PO
DIRBLKS: 10
RECFM: FB
LRECL: 80
BLKSIZE: 23440 (or other valid blocksize)
SPACE: 2 tracks
Usage: Sample LINK and execute JCL

DSNAME: hlq1.hlq2.MSGCAT
DSORG: PS
DIRBLKS: 10
RECFM: FB
LRECL: 100
BLKSIZE: 32000 (or other valid blocksize)
SPACE: 2 tracks
Usage: Message text

DSNAME: hlq1.hlq2.OBJECT
DSORG: PO
DIRBLKS: 10
RECFM: FB
```

```
LRECL: 80
BLKSIZE: 3200
SPACE: 1 cylinder
Usage: Pre-linked object files

DSNAME: hlq1.hlq2.SOURCE
DSORG: PO
DIRBLKS: 10
RECFM: FB
LRECL: 80
BLKSIZE: 23440 (or other valid blocksize)
SPACE: 1 track
Usage: Sample IMS client and server transactions
```

The directory structures on the eLink TCP for IMS software CD include the following runtime datasets.

Runtime Datasets	Format	Contents
LOAD	ASCII	Load library for BEA eLink TCP for IMS
CONFIG	ASCII	Configuration file for BEA eLink TCP for IMS
MSGLOG	ASCII	Message log for BEA eLink TCP for IMS
SVRLOG	ASCII	A log file containing server responses for BEA eLink TCP for IMS

Allocate the following runtime datasets for use by the eLink TCP for IMS (*hlq1* and *hlq2* are dataset name high-level qualifiers appropriate to your installation).

Listing 2-2 Allocating Runtime Data Sets

```
DSNAME: hlq1.hlq2.LOAD
DSORG: PO
DIRBLKS: 10
RECFM: U
```

```
LRECL: 0
BLKSIZE: 6133 (or other valid blocksize)
SPACE: 1 cylinder
Usage: Load library

DSNAME: hlq1.hlq2.CONFIG
DSORG: PS
RECFM: FB
LRECL: 80
BLKSIZE: 23440 (or other valid blocksize)
SPACE: 1 track
Usage: Configuration File

DSNAME: hlq1.hlq2.MSGLOG
DSORG: PS
RECFM: VB
LRECL: 1028
BLKSIZE: 6144 (or other valid blocksize)
SPACE: 1 cylinder
Usage: Message Log

DSNAME: hlq1.hlq2.SVRLOG
DSORG: PS
RECFM: VB
LRECL: 32756
BLKSIZE: 32760 (Note: required blocksize)
SPACE: 1 cylinder
Usage: Server Response Log
```

Unload the Distribution Files

The eLink TCP for IMS product is distributed on a single CD-ROM (along with other members of the BEA eLink product family). The eLink TCP for IMS distribution files are grouped under one of the following directory structures:

- For UNIX the directory is `/cdrom/unixmf` where `/cdrom` is the directory where the drive is mounted.
- For Windows NT the directory is `D:\ntmf` where `D:` is the CD-ROM drive.

Because the distribution files are located on a CD-ROM, you probably use a UNIX-based system or a PC-based system to access the files on the CD-ROM and file transfer them to OS/390.

You must transfer the contents of each subdirectory to the corresponding distribution library on the OS/390 system. You can do this by using a file transfer program (such as FTP) or any other mechanism that allows you to transfer files to your OS/390 system. The file transfer program must have the following capabilities:

- Properly transferring both text and binary files
- Properly creating members in a target library (PDS)

The IBM File Transfer Program (FTP, distributed as part of TCP/IP for OS/390) satisfies both of these requirements.

Warning: If you are running IBM TCP 3.1, you *must* include the following lines of code when using FTP to transfer files. If you do not, the files are corrupted during the transfer.

If you are running a version of TCP/IP other than IBM TCP 3.1, the following lines of code do not impact the transfer. An error message may result that can be ignored.

Listing 2-3 FTP using IBM TCP 3.1

```
ftp> quot site rec=fb
ftp> quot site lr=80
```

Each file in each subdirectory on the distribution CD-ROM corresponds to a member of the associated target library. For example, file BEATCPI in the JCL subdirectory corresponds to member BEATCPI in the JCL distribution library.

Note: All files in the CONTROL, INCLUDE, JCL, and SOURCE subdirectories are *text files*; be sure to specify *text (or ASCII) mode* when you transfer these files from the CD-ROM to their corresponding OS/390 distribution library members.

The MSGCAT file and all files in the OBJECT subdirectory are *binary files*; you must specify *binary (or image) mode* when you transfer these files from the CD-ROM to the corresponding OS/390 distribution library members.

Link-Edit eLink TCP for IMS

The eLink TCP for IMS program is delivered as a pre-linked object file.

- Use the LNKBEA31 JCL file for linking with IBMs TCP/IP V3R1 product.
- Use the LNKBEAIL file for linking Interlink's TCPAccess Version 4.1 product.
- Use the LNKBEA34 JCL file for linking with IBM's TCP/IP V3R2 or V3R4 products.

You need to tailor the JCL as required to match your installation's standards and naming conventions. Review the following items and make the necessary changes before you submit the job.

1. Supply an appropriate Job Card for the job.
2. Change the names of product datasets as required to match those in your particular installation, including LE/370 datasets (for example, SYS1.SCEELKED), TCP/IP for OS/390 datasets (for example, SYS1.SEZACMTX), Interlink datasets (for example, SYS1.TCPACC.CLIB), and IMS (for example, IMSV5R1.RESLIB).
3. Change the names of eLink TCP for IMS datasets as required to match those you allocated in Step 1 (for example, BEATCPI.DIST.OBJECT).
4. Change the name of the link-editor (HEWL) as required to match the name used by your installation.

Note: The link job references components supplied by LE/370, TCP/IP for OS/390, and IMS.

When the job completes successfully, you receive a return code of 0 from the link-edit steps and an executable load module (the eLink TCP for IMS BMP) is placed into the specified target load library. The load module should have attributes of AMODE=31, RMODE=24. There should be no unresolved external references.

Sample JCL for Linking eLink TCP for IMS

Listing 2-4 is JCL to link the distributed object files into an executable load module.

Listing 2-4 Sample JCL for Linking Distributed Object Files

```
//JOBNAME JOB ETC.
//LKED    EXEC PGM=HEWL,COND=(4,LT),
//       REGION=2M,PARM='AMODE=31,MAP,XREF'
//SYSLIB  DD DSNAME=SYS1.SCEELKED,DISP=SHR
//       DD DSNAME=SYS1.SEZACMTX,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSLMOD DD DISP=SHR,DSNAME=BEATCPI.DIST.LOAD
//SYSUT1  DD UNIT=VIO,SPACE=(TRK,(10,10))
//RESLIB  DD DISP=SHR,DSN=IMSV5R1.RESLIB
//SYSLIN  DD DISP=OLD,DSN=&&PRELINK
//        DD *
 INCLUDE RESLIB(DFSLI000)
 ENTRY CEESTART
 NAME BEATCPI(R)
/*
//
```

The IBM prelinker must be used to prelink the object modules before invoking the linkage editor.

Define eLink TCP for IMS to IMS

To integrate eLink TCP for IMS into your IMS environment, you must properly define it to IMS.

BMP Definition

If using the BMP, to integrate eLink TCP for IMS into your IMS environment, you must properly define it to IMS. Member BEATCPI in the CONTROL distribution library contains the required IMS definitions, including the appropriate APPLCTN, TRANSACT, PCB, and PSBGEN macros. These definitions should be furnished to an IMS systems programmer who can include them in the overall IMS system definition. The APPLCTN and TRANSACT definitions are included as part of the input to an IMS Stage 1 gen, while the PCB and PSBGEN definitions are processed as input to a subsequent PSB gen.

Use the definitions as supplied; *do not make changes*. In particular, note the following:

- BEA eLink TCP for IMS is a transaction-oriented BMP.
- BEA eLink TCP for IMS is *not* a WFI (wait for input) BMP; do *not* attempt to define it as such.
- EXPRESS=YES must be specified.
- BEA eLink TCP for IMS must have a modifiable Alternate PCB.
- BEA eLink TCP for IMS is written in the C programming language; therefore, LANG=ASSEM must be specified (or allowed to default).

Note: If desired, you *may* change the BMP transaction code and/or PSB name (the supplied default is BEATCPI). If you do, however, be sure to make the corresponding changes elsewhere (for example, in your user-written Client and Server transactions).

Once the required Stage 1 and PSB gens have been successfully completed, BEA eLink TCP for IMS should be defined to IMS.

Any BEA eLink TCP for IMS Client and Server transactions that you write must also be properly defined to IMS. In general, BEA eLink TCP for IMS imposes no special requirements on these definitions, so you define them exactly as you would any IMS transaction. In order to communicate with BEA eLink TCP for IMS through the IMS Message Queue, however, user-written Client and Server transactions must be defined with a modifiable Alternate PCB.

OTMA Definition

If using OTMA, to integrate eLink TCP for IMS into your OS/390 environment, you must properly define it in the program property table (PPT). Listing 2-5 is a sample program entry for the eLink TCP product. Also, you must link the program to an APF authorized library.

Listing 2-5 Sample Program Property Table

```
PPT PGMNAME(BEATCP34)     /* PROGRAM NAME = BEATCP34           */
        CANCEL            /* PROGRAM CAN BE CANCELED           */
        KEY(7)            /* PROTECT KEY ASSIGNED IS 7         */
        SWAP              /* PROGRAM IS SWAPPABLE              */
        NOPRIV            /* PROGRAM IS NOT PRIVILEGED         */
        DSI               /* REQUIRES DATA SET INTEGRITY       */
        PASS              /* CANNOT BYPASS PASSWORD PROTECTION */
        SYST              /* PROGRAM IS A SYSTEM TASK          */
        AFF(NONE)         /* NO CPU AFFINITY                   */
        NOPREF            /* NO PREFERRED STORAGE FRAMES       */
```

Create an eLink TCP for IMS Configuration File

BEA eLink TCP for IMS makes use of a text configuration file which is read and processed during initialization to establish the configuration. Any text editor (such as the ISPF Editor) can be used to create the configuration file. Members CONFIGOT (OTMA sample) and CONFIGBP (BMP sample) in the CONTROL distribution library are sample configuration files which you may use as a starting point for creating configuration files tailored to your particular installation. Place your completed configuration file in the configuration dataset allocated in the “Allocate OS/390 Data Sets” section.

A configuration file includes the following basic sections:

- Define SYSTEM Statement
- Define Local Gateways
- Define Remote Gateways
- Define Remote Services
- Define Local Services and IMS Host Systems

  Note: You must define local services and IMS host systems if running OTMA.

For detailed information about syntax and parameter definitions for the configuration file, refer to the *BEA eLink TCP for IMS User Guide*.

Define SYSTEM Statement

SYSTEM parameters control the overall operational environment of BEA eLink TCP for IMS. Because eLink TCP for IMS cannot "wait" on the IMS Message Queue, the queue is periodically interrogated (polled) to determine if any new work has arrived (i.e., IMS Client requests or Server responses). The SleepTime parameter specifies the amount of time that eLink TCP for IMS "sleeps" (waits) during idle periods before checking the IMS Message Queue for work.

Define Local Gateways

You must provide at least one GATEWAY TYPE=LOCAL statement to define access to your local (IMS) gateway by remote systems. In particular, this statement specifies:

- The IP Address that remote systems use to establish TCP/IP connections with this gateway

  The IP Address should be that of the OS/390 machine on which BEA eLink TCP for IMS is running.

- The Port Number that remote systems use to establish TCP/IP connections with this gateway

You should specify a Port Number that is not a well-known port number and one that is not used by any other TCP/IP application.

- The Account ID and Password (if used) that must be supplied by remote systems to successfully establish an inbound session with this gateway
- The maximum number of concurrent inbound sessions (TCP/IP connections initiated by remote systems) that is accepted via the specified port

Make sure that remote systems are correctly configured with the corresponding IP Address, Port Number, Account ID (if used), and Password (if used). Otherwise, attempts by remote systems to connect to this gateway fail.

Define Remote Gateways

GATEWAY TYPE=REMOTE statements define access to remote systems by the IMS gateway. In particular, each GATEWAY TYPE=REMOTE statement specifies:

- The IP Address to use to establish a TCP/IP connection (that is, connect) with the remote gateway
- The Port Number to use to establish a TCP/IP connection (that is, connect) with the remote gateway
- The Account ID and Password (if used) that must be supplied to establish an outbound session with the remote gateway
- The minimum and maximum number of concurrent, outbound sessions that are established with the remote gateway

  If you specify a value other than zero for the minimum number of sessions, eLink TCP for IMS attempts to establish the specified number of outbound sessions with the remote system during initialization.

The IP Address, Port Number, Account ID, and Password must match the configuration of the remote system. Otherwise, attempts to establish an outbound session with the remote system fails.

Define Remote Services

SERVICE TYPE=REMOTE statements define remote services to which BEA eLink TCP for IMS has access. Each statement specifies:

- The corresponding local and remote service names

  The Local Service Name is the name used by an IMS Client transaction to request the service. The Remote Service Name is the name of the service as defined on the remote system. These need not be the same, but note that the Remote Service Name *must* match the configuration of the remote system.

- The remote system (Logical Machine ID) that offers the service

  The remote system offering the service is identified by a Logical Machine ID, which must correspond to the Logical Machine ID specified in a GATEWAY TYPE=REMOTE statement.

- A time limit for processing the request

  The Time Limit (in seconds) specifies how long BEA eLink TCP for IMS "waits" for a response to a request sent to a remote system before assuming that the request has "timed out" and that no response is forthcoming. Specify a value that makes a reasonable allowance for network latency and overall system load. Small values should be avoided because they may result in requests timing out under otherwise normal circumstances.

Note: It is permissible to have more than one SERVICE TYPE=REMOTE statement for the same (local) service name. When this is the case, each statement must identify a different remote Logical Machine ID. This allows BEA eLink TCP for IMS to "spread" requests for a particular service among multiple remote systems offering the service, thus balancing the load and improving overall throughput.

Define Local Services and IMS Host Systems

SERVICE TYPE=LOCAL statements define local services offered by a local IMS host. There can be multiple SERVICE TYPE=LOCAL statements and multiple statements defining the same service, provided each specifies a different local IMSID. Each statement requires the following information:

- The corresponding local and remote service names

  The Local Service Name is the name used by an IMS Client transaction to request the service. The Remote Service Name is the name of the service as defined on the remote system.

  Note: The local and remote service names can be different; however, the Remote Service Name *must* match the configuration of the remote system.

- The IMSID (Logical IMS ID) that offers the service

  The IMSID is a symbolic name that uniquely identifies an IMS system definition. This name must match a logical IMS ID (IMSID) specified in a HOST, TYPE=IMS statement.

Create JCL to Run eLink TCP for IMS

You must create JCL to run the BEA eLink TCP for IMS product. The following sections specify how to create JCL for running as a BMP or as an OTMA client.

JCL for BMP

When running the product as a BMP, BEA eLink TCP for IMS is a standard, transaction-oriented IMS BMP (Batch Message Processing) program. IMS BMPs are submitted and run as ordinary batch jobs.

IMS normally supplies a cataloged procedure (PROC) for executing BMPs, and you should use the procedure supplied with your release of IMS, if available. Member RUNBMP in the JCL distribution library is a sample job for executing eLink TCP for IMS and can be used as a reference when modifying the JCL supplied with your version of IMS.

When tailoring the JCL for your BMP installation, pay particular attention to the following:

- You need to supply an appropriate Job Card for the job. The job card should specify TIME=1440, because eLink TCP for IMS executes as a non-ending job.

- BEA eLink TCP for IMS is written in the C language and requires access to the LE/370 runtime environment. Add DD cards to the STEPLIB for the LE/370 runtime datasets, including SYS1.SCEERUN and SYS1.SCEELKED. Use whatever dataset names are appropriate for your installation.
- For diagnostic purposes, you may wish to add DD cards for SYSUDUMP and/or CEEDUMP. In the event of an abnormal termination (ABEND), LE/370 writes a formatted dump to CEEDUMP.
- Add a DD card for SYSTCPD pointing to the TPC/IP Data file (e.g., TCPIP.V3R1.DATA – change the dataset name as necessary to conform to your installation). This dataset is used by TCP/IP in the BEA eLink TCP for IMS address space (when using IBM TCP/IP).
- Add DD cards for the eLink TCP for IMS runtime datasets: CONFIG, MSGLOG, MSGCAT, and SVRLOG (all of which were allocated in Step 1, above).
- If you want to preserve messages from previous executions of eLink TCP for IMS, code DISP=MOD in the MSGLOG DD statement; new messages are appended to the end of the dataset (existing messages are preserved). If you want the Message Log to be overwritten with each new execution of BEA eLink TCP for IMS, code DISP=OLD (or DISP=SHR) in the MSGLOG DD statement; existing messages are lost.
- Code CKPTID=NOMSG681 on the BMP proc invocation to suppress IMS message DFS681I (which would otherwise be issued every time the BMP issues a CHKP call).
- Code PGMLIB=*hlq1.hlq2*.LOAD on the BMP proc invocation to identify the library containing the BMP (eLink TCP for IMS).
- Code the proper values for the MBR, IN, and PSB parameters on the BMP proc invocation. The default for each of these is BEATCPI, but if you elect to change the load module name, transaction code, or PSB, code the appropriate name(s).

JCL for OTMA Clients

When running the product as an OTMA client, BEA eLink TCP for IMS runs in its own address space under OS/390 as a started task or long running job. Member RUNOTM in the JCL distribution library is a sample job for executing BEA eLink TCP for IMS.

When tailoring the JCL for your OTMA installation, pay particular attention to the following:

- You need to supply an appropriate Job Card for the job. The job card should specify TIME=1440, since BEA eLink TCP for IMS executes as a non-ending job.
- BEA eLink TCP for IMS is written in the C language and requires access to the LE/370 runtime environment. Add DD cards to the STEPLIB for the LE/370 runtime datasets, including SYS1.SCEERUN and SYS1.SCEELKED. Use whatever dataset names are appropriate for your installation.
- For diagnostic purposes, you may wish to add DD cards for SYSUDUMP and/or CEEDUMP. In the event of an abnormal termination (ABEND), LE/370 writes a formatted dump to CEEDUMP.
- Add a DD card for SYSTCPD pointing to the TPC/IP Data file (e.g., TCPIP.V3R1.DATA – change the dataset name as necessary to conform to your installation). This dataset is used by TCP/IP in the BEA eLink TCP for IMS address space (when using IBM TCP/IP).
- If you want to preserve messages from previous executions of BEA eLink TCP for IMS, code DISP=MOD in the MSGLOG DD statement; new messages are appended to the end of the dataset (existing messages are preserved). If you want the Message Log to be overwritten with each new execution of BEA eLink TCP for IMS, code DISP=OLD (or DISP=SHR) in the MSGLOG DD statement; existing messages are lost.

Start BEA eLink TCP for IMS

Before you attempt to start eLink TCP for IMS as a BMP for the first time, make sure that the proper IMS definitions are in place and that the eLink TCP for IMS transaction code and program specification block (PSB) are defined and started. If the PSB is stopped, the job immediately abends.

If you are using RACF for security, you must set up a FACILITY Class Profile in RACF of IMSXCF.*group.client*. You can configure the *group* and *client* names for BEA eLink TCP for IMS.

Start BEA eLink TCP for IMS by submitting the JCL prepared in the "Create JCL to Run eLink TCP for IMS" section.

For additional information on operating this product, refer to the *BEA eLink TCP for IMS User Guide*.

Test the Installation

When you start BEA eLink TCP for IMS for the first time, you should do so in a controlled environment, using a small, simple configuration to methodically test the system to verify your installation and the configuration. For additional information on using this product, refer to the *BEA eLink TCP for IMS User Guide*.

3 Installing BEA eLink TCP for CICS

Installing BEA eLink for Mainframe TCP for CICS (hereafter referenced as eLink TCP for CICS) consists of the following basic tasks and topics.

1. Verify System Requirements
2. Pre-Installation Considerations
3. Allocate Temporary Data Sets
4. Transferring the Installation JCL
5. Unloading the Distribution Files
6. Define the CICS VSAM Files
7. Verify Contents of PROCS
8. Changing Resource Names (Optional)
9. Define the CICS Table Entries
10. Build eLink TCP for CICS Executables
11. Verify the CICS Setup
12. Assemble and Link the CICS DCT
13. Modify the CICS Startup JCL

Verify System Requirements

Prior to beginning the installation, verify hardware and software requirements. For system requirements, refer to the *BEA eLink TCP Release Notes*. Ensure that your installation workstation is connected to OS/390 through TCP/IP, and is capable of running FTP.

Pre-Installation Considerations

Verify that all block sizes in the supplied JCL are appropriate for the device types being used. By default, 3380 is assumed.

Allocate Temporary Data Sets

Allocate a temporary data set on OS/390 direct access storage device (DASD) for the installation job as shown in the following example.

Note: You can specify either the Volume Serial parameter or the Generic Unit parameter, but not both.

Listing 3-1 Allocating Data Sets

```
Data Set Name:YOURHLQ.TEMP.INSTALL
Volume serial:nnnnn         (Blank for authorized default volume)
Generic unit:               (Generic group name or unit address)
Space units:  TRACK         (BLKS, TRKS, CYLS, KB, MB or BYTES)
Primary quantity:1          (In above units)
Secondary quantity:1        (In above units)
Directory blocks:0          (Zero for sequential data set)
Record format:FB
Record length:80
```

```
Block size:6160
Expiration date:            (YY/MM/DD, YYYY/MM/DD YY.DDD,
                            YYYY.DDD in Julian form DDDD for
                            retention period in days or blank)
Enter "/" to select option
Allocate Multiple Volumes
```

Transferring the Installation JCL

The eLink TCP for CICS product is distributed on a single CD-ROM (with other members of the BEA eLink TCP product family). The eLink TCP for CICS distribution files are grouped under one of the following directory structures:

- To FTP from UNIX to the mainframe, the distribution files are in the `/cdrom/unixmf/cics/v3x` directory, where `/cdrom` is the directory where the drive is mounted and `v3x` is the directory specifying the IBM TCP/IP version. Select one of the following directories depending on the TCP/IP stack you are using.
  - For Interlink TCPaccess 4.1, use `/cdrom/unixmf/cics/v31`
  - For IBM TCP/IP 3.1, use `/cdrom/unixmf/cics/v31`
  - For IBM TCP/IP 3.2 or higher, use `/cdrom/unixmf/cics/v32`
- To FTP from Windows NT to the mainframe, the distribution files are in the directory `D:\ntmf\cics\v3x,` where `D:` is the CD-ROM drive and `v3x` is the directory specifying the IBM TCP/IP version. Select one of the following directories depending on the TCP/IP stack you are using.
  - For Interlink TCPaccess 4.1, use `D:\ntmf\cics\v31`
  - For IBM TCP/IP 3.1, use `D:\ntmf\cics\v31`
  - For IBM TCP/IP 3.2 or higher, use `D:\ntmf\cics\v32`

FTP the file called INSTALL to the temporary data set created in the “Allocate Temporary Data Sets” section. Listing 3-3 is the process for unloading the distribution files using FTP.

Warning: If you are running IBM TCP 3.1, you *must* include the following lines of code when using FTP to transfer files. If you do not, the files are corrupted during the transfer.

If you are running a version of TCP/IP other than IBM TCP 3.1, the following lines of code do not impact the transfer. An error message may result that can be ignored.

Listing 3-2 FTP using IBM TCP 3.1

```
ftp> quot site rec=fb
ftp> quot site lr=80
```

Listing 3-3 Unloading Distribution Files

```
% ftp os390-node-name
Connected to os390-node-name.
Name (os390-node-name:unix-id): tso-id
331 Send password please.
Password: tso-password
230 tso-id is logged on.
Remote system type is OS/390.
ftp> quot site rec=fb
ftp> quot site lr=80
ftp> put INSTALL 'dataset name allocated'
200 Port request OK.
125 Storing data set 'dataset name allocated'
2749 bytes sent in 0.00 seconds (555.12 Kbytes/s)
ftp> quit
221 Quit command received. Goodbye.
```

Performing the Installation

You need to tailor the JCL as required to match your installation's standards and naming conventions. Review the following items and make the necessary changes before you submit the job.

Note: This is an instream PROC; the variables are at the end of the PROC.

1. Modify the JOBCARD to your standards.
2. Modify the variable “YOURHLQ.” If you intend to rename them, modify the variables “JCL,” “OBJECT,” and “SOURCE.”
3. Submit the revised temporary data. This JCL creates six partitioned data sets based on the “high-level-qualifier” you have specified as the value of the JCL parameter (&HLQ) in the JCL.

The INSTALL JCL includes the name “BEATCPC” as part of the data set name. For example.

Listing 3-4 Sample INSTALL JCL

```
HLQ='YOURHLQ'

"YOURHLQ".BEATCPC.CONTROL
"YOURHLQ".BEATCPC.JCL
"YOURHLQ".BEATCPC.OBJECT
"YOURHLQ".BEATCPC.SOURCE
"YOURHLQ".BEATCPC.INCLUDE
"YOURHLQ".BEATCPC.PLKED
```

Unloading the Distribution Files

Unload all the files on the distribution medium to the partitioned data sets allocated by the job INSTALL in the “Performing the Installation” section.

The eLink TCP for CICS distribution files are grouped under one of the following directory structures:

- To FTP the distribution files from UNIX to the mainframe, the files are in the directory `/cdrom/unixmf/cics/v3x`, where `/cdrom` is the directory where the drive is mounted and `v3x` is the directory specifying the IBM TCP/IP version. Select one of the following directories depending on the TCP/IP stack you are using.

- For Interlink TCPaccess 4.1, use `/cdrom/unixmf/cics/v31`
- For IBM TCP/IP 3.1, use `/cdrom/unixmf/cics/v31`
- For IBM TCP/IP 3.2 or higher, use `/cdrom/unixmf/cics/v32`

- To FTP files from Windows NT to the mainframe, the files are in the directory `D:\ntmf\cics\v3x`, where `D:` is the CD-ROM drive and `v3x` is the directory specifying the IBM TCP/IP version. Select one of the following directories depending on the TCP/IP stack you are using.
  - For Interlink TCPaccess 4.1, use `D:\ntmf\cics\v31`
  - For IBM TCP/IP 3.1, use `D:\ntmf\cics\v31`
  - For IBM TCP/IP 3.2 or higher, use `D:\ntmf\cics\v32`

The previous directory structures include the following partitioned data sets.

From	To
`./control`	Your PDS for CONTROL (YOURHLQ.BEATCPC.CONTROL) (ASCII mode)
`./jcl`	Your PDS for JCL (YOURHLQ.BEATCPC.JCL) (ASCII mode)
`./source`	Your PDS for SOURCE (YOURHLQ.BEATCPC.SOURCE) (ASCII mode)
`./include`	Your PDS for INCLUDE (YOURHLQ.BEATCPC.INCLUDE) (ASCII mode)
`./object`	Your PDS for OBJECT (YOURHLQ.BEATCPC.OBJECT) (Binary mode)
`./plked`	Your PDS for PLKED (YOURHLQ.BEATCPC.PLKED) (Binary mode)

Warning: If you are running IBM TCP 3.1, you *must* include the following lines of code when using FTP to transfer files. If you do not, the files are corrupted during the transfer.

If you are running a version of TCP/IP other than IBM TCP 3.1, the following lines of code do not impact the transfer. An error message may result that can be ignored.

Listing 3-5 FTP using IBM TCP 3.1

```
ftp> quot site rec=fb
ftp> quot site lr=80
```

Listing 3-6 Unloading Distribution Files Using FTP

```
% ftp os390-node-name
Connected to os390-node-name.
Name (beavs:tso-id):
331 Send password please.
Password: tso-password
230 tso-id is logged on.
Remote system type is OS/390.
ftp> prompt off
ftp> quot site rec=fb
ftp> quot site lr=80
Interactive mode off.
```

Upload the `control` files.

Note: You must set ASCII mode on.

Listing 3-7 Upload control Files

```
ftp> lcd /cdrom/unixmf/cics/control
Local directory now /cdrom/unixmf/cics/control
ftp> cd 'YOURHLQ.BEATCPC.CONTROL'
250 "'YOURHLQ.BEATCPC.CONTROL'" partitioned data set is working
directory
ftp> mput *
local: CSDU remote: CSDU
200 Port request OK.
```

```
125 Storing data set "YOURHLQ".BEATCPC.CONTROL(CSDU)
250 Transfer completed successfully.
5740 bytes sent in 0.01 seconds (747.10 Kbytes/s)
.
.
.
```

Upload the `jcl` files.

Note: You must set ASCII mode on.

Listing 3-8 Uploading jcl Files

```
ftp> lcd /cdrom/unixmf/cics/jcl
Local directory now /cdrom/unixmf/cics/jcl
ftp> cd 'YOURHLQ.BEATCPC.JCL'
250 "'YOURHLQ.BEATCPC.JCL'" partitioned data set is working
directory.
ftp> mput *
local: BLDVSAM remote: BLDVSAM
200 Port request OK.
125 Storing data set "YOURHLQ".BEATCPC.JCL(BLDVSAM)
250 Transfer completed successfully.
2214 bytes sent in 0.00 seconds (563.49 Kbytes/s)
.
.
.
```

Upload the `source` files.

Note: You must set ASCII mode on.

Listing 3-9 Uploading source Files

```
ftp> lcd /cdrom/unixmf/cics/source
Local directory now /cdrom/unixmf/cics/source
ftp> cd 'YOURHLQ.BEATCPC.SOURCE'
250 "'YOURHLQ.BEATCPC.SOURCE'" partitioned data set is working
directory.
ftp> mput *
local: BEACCLN1 remote: BEACCLN1
200 Port request OK.
125 Storing data set YOURLHQ.BEATCPC.SOURCE(BEACCLN1)
250 Transfer completed successfully.
10578 bytes sent in 0.01 seconds (1013.85 Kbytes/s)
.
.
.
```

Upload the `include` files.

Note: You must set ASCII mode on.

Listing 3-10 Uploading include Files

```
ftp> ascii
ftp> lcd /cdrom/unixmf/cics/include
ftp> cd '"YOURHLQ".BEATCPC.INCLUDE'
250 "'YOURHLQ.BEATCPC.INLUDE'" partitioned data set is working
directory
ftp> mput *
200 Port request OK.
125 Storing data set "YOURHLQ".BEATCPC.INCLUDE'(CLIENT)
250 Transfer completed successfully.
local: CLIENT remote: CLIENT
3736 bytes sent in 0.0081 seconds (4.5e+02 Kbytes/s)
.
.
.
```

Upload the `object` files.

Note: You must set binary mode on.

Listing 3-11 Uploading object Files

```
ftp> binary
200 Representation type is IMAGE.
ftp> lcd /cdrom/unixmf/cics/object
Local directory now /cdrom/unixmf/cics/object
ftp> cd 'YOURHLQ.BEATCPC.OBJECT'
250 "'YOURHLQ.BEATCPC.OBJECT'" partitioned data set is working
directory.
ftp> mput *
local: BEACFGSV remote: BEACFGSV
200 Port request OK.
125 Storing data set "YOURHLQ".BEATCPC.OBJECT(BEACFGSV)
250 Transfer completed successfully.
6240 bytes sent in 0.00 seconds (1466.96 Kbytes/s)
.
.
.
```

Upload the `plked` files.

Note: You must set binary mode on.

Listing 3-12 Uploading plked Files

```
ftp> binary
200 Representation type is Image
ftp> lcd /cdrom/unixmf/plked
ftp> cd '"YOURHLQ".BEATCPC.PLKED'
250 "'"YOURHLQ.BEATCPC.PLKED'" partitioned data set is working
directory
ftp> mput *
200 Port request OK.
125 Storing data set "YOURHLQ.BEATCPC.PLKED(BEACIC00)
250 Transfer completed successfully.
```

```
local: BEACIC00 remote: BEACIC00
64560 bytes sent in 0.7 seconds (90 Kbytes/s)
.
.
.
```

Verify that the distribution libraries were unloaded. For the listing of files that should be on your system, refer to the “List of Distribution Files and Members” section.

Define the CICS VSAM Files

Define the CICS VSAM Files (Connections, Requesters, Service Names, User Connection, Inbound Service).

1. Modify DELETE/DEFINE control cards VCON, VREQ, VSVC, VUSR, and VISN in the PDS "YOURHLQ".BEATCPC.CONTROL. Listing 3-13 is an example of a modified DELETE/DEFINE.

Listing 3-13 Sample DELETE/DEFINE Control Card

```
DELETE YOURHLQ.BEATCPC.BEAVSVC                          -
        CLUSTER PURGE                                   -
DEFINE CLUSTER (                                        -
        NAME (YOURHLQ.BEATCPC.BEAVSVC)                  -
        SHAREOPTIONS (2 3)                              -
        KEYS (16 0)                                     -
        CISZ (1024)                                     -
        TRACKS (5 1)                                    -
        VOLUMES (CICS01))                               -
       DATA (                                           -
        RECORDSIZE (56, 56)                             -
        NAME (YOURHLQ.BEATCPC.BEAVSVC.DATA))            -
       INDEX (                                          _
        NAME (YOURHLQ.BEATCPC.BEAVSVC.INDEX))
```

Change all instances of "YOURHLQ.BEATCPC.BEAVSVC" to a physical data set name that meets your standards. Change the TRACKS and VOLUMES parameters to appropriate values for your installation. The provided value for the TRACKS parameter works for most installations, but this value can be increased if necessary.

2. For the BLDVSAM JCL, complete the following steps.

   a. Modify the JOBCARD to your standards.

   b. Modify "YOURHLQ.BEATCPC.XXXX", where XXXX is BEAVCON, BEAVREQ, etc.

   c. Submit BLDVSAM.

3. Verify the results. There should be three data sets in "YOURHLQ.BEATCPC" corresponding to each of the five VSAM files. The three data sets are as follows:

   - One data set has no extension
   - One data set has DATA as the extension
   - One data set has INDEX as the extension

   Note: A condition code of 8 is acceptable on DELETE steps. If you receive condition codes other than zero for DEFINE steps, you should evaluate this for potential problems.

Verify Contents of PROCS

Before you go on to the next step, we recommend that you check the following items in PROCS CMPBC, CMPLMAP, CMPLKED, LNKIBM, and LNKINT in the PDS "YOURHLQ".BEATCPC.JCL".

You may need to modify them to meet your standards.

- STEPLIB after the EXEC PGM=
- SYSMSGS after the EXEC PGM=
- SYSLIB after the EXEC PGM=

- UNIT=WORK

Changing Resource Names (Optional)

The default names for BEA eLink TCP for CICS Mapsets, Programs, Transids, and VSAMs are listed in Appendix A, “Changing Resource Names.” If you want to use a different naming convention, use the procedure for changing names included after each table.

Note: You must have the C 370 compiler to customize resource names.

Define the CICS Table Entries

Verify the CICS table entry definitions (PCTs, PPTs, FCTs) in CSDU (from the PDS "YOURHLQ".BEATCPC.CONTROL).

Modify the group and list name in CSDU (from the PDS "YOURHLQ".BEATCPC.CONTROL).

In the JCL in CSDUPD (from the PDS "YOURHLQ".BEATCPC.JCL), complete the following steps.

1. Modify the JOBCARD to your standards.
2. Modify the SYSIN DSN=YOURHLQ.BEATPC.CONTROL.
3. Submit CSDUPD.
4. Verify the results.

Build eLink TCP for CICS Executables

To build the executables for BEA eLink TCP for CICS, you need to complete the following tasks.

1. If there is not a PDS allocated where the executables for your CICS region reside, allocate such a PDS before running any JCL that builds executables.
2. Link the CICS Admin maps (MainMenu, Connection, Requester, Outbound Service Name, Inbound Service Name, User Connection Account, Handler Configuration). In the JCL in MAKLMAP (from the PDS "YOURHLQ".BEATCPC.JCL"), complete the following steps.
   a. Modify the JOBCARD to your standards.
   b. Modify the PROCS JCLLIB card with the correct HLQ.
   c. Modify the variables YOURHLQ, PDSOBJS and LNKLIB for your PDS.

   Note: LNKLIB is where your CICS region executable binary files reside.

   d. Modify the variable PROG, if you renamed mapsets.
   e. Submit MAKLMAP.
   f. Verify the results.
3. Complete one of the following procedures depending on your compiler:
   - Pre-Link/Link eLink TCP for CICS
   - Linking without the C 370 Compiler

Pre-Link/Link eLink TCP for CICS

The following sections explain the procedures for linking programs and mapsets prior to running the BEA eLink TCP for CICS product.

Note: The following sections apply for a C 370 compiler.

Linking the CICS Admin Programs

Pre-Link/Link the CICS Admin Pgms (MainMenu, Connection, Requester, Outbound Service Name, Inbound Service Name, User Connection Account, Handler Configuration). In the JCL in MAKL1 (from the PDS "YOURHLQ".BEATCPC.JCL"), complete the following steps.

1. Modify the JOBCARD to your standards.
2. Modify the PROCS JCLLIB card with the correct HLQ.
3. Modify the variables YOURHLQ, PDSOBJS, TCPLIB and LNKLIB for your PDS.

   Note: LNKLIB is where your CICS region executable binary files reside.
4. Modify the SYSIN NAME Pgm(R), if you renamed programs.
5. Submit MAKL1.
6. Verify the results.

Linking the CICS Utility Programs

Pre-Link/Link the CICS utility Pgms (Pre-Requester, Application Handler, Shutdown, and Configuration Converter). In the JCL in MAKL0 (from the PDS "YOURHLQ".BEATCPC.JCL"), complete the following steps.

1. Modify the JOBCARD to your standards.
2. Modify the PROCS JCLLIB card with the correct HLQ.
3. Modify the variables YOURHLQ, PDSOBJS, TCPLIB and LNKLIB for your PDS.

   Note: LNKLIB is where your CICS region executable binary files reside.
4. Modify the SYSIN NAME Pgm(R), if you renamed programs.
5. Submit MAKL0.
6. Verify the results.

Linking the CICS TCP Programs

Pre-Link/Link the CICS TCP Pgms (Requester and Handler). Proceed to the appropriate section depending on the TCP/IP stack you are running.

- Interlink CPT
- IBM TCP/IP

Interlink CPT

In the JCL in MAKL2 (from the PDS "YOURHLQ".BEATCPC.JCL"), complete the following steps.

1. Modify the JOBCARD to your standards.
2. Modify the variables YOURHLQ, PDSOBJS and LNKLIB for your PDS.

   Note: LNKLIB is where your CICS region executable binary files reside.
3. Modify the SYSIN NAME Pgm(R), if you renamed programs.
4. Submit MAKL2.
5. Verify the results.

Note: A sample entry for the configuration file for INTERLINK is located in `"YOURHLQ".BEATCPC.SOURCE(T09CONFG)`.

IBM TCP/IP

In the JCL in MAKL3 (from the PDS "YOURHLQ".BEATCPC.JCL"), complete the following steps.

1. Modify the JOBCARD to your standards.
2. Modify the variables YOURHLQ, PDSOBJS, TCPLIB and LNKLIB for your PDS.

   Note: LNKLIB is where your CICS region executable binary file resides.
3. Modify the INCLUDE OBJECT(Pgm), if you renamed programs.
4. Modify the SYSIN NAME Pgm(R), if you renamed programs.
5. Submit MAKL3.

6. Verify the results.

Linking without the C 370 Compiler

If you do NOT have the C 370 compiler, use the following procedure to Linkedit the PreLinked objects. Link the Prelinked objects of the following programs.

- CICS TCP Pgms (Requester and Handler).
- CICS Utility Pgms (Pre-Requester, Application Handler, Shutdown, and Configuration Converter).
- CICS Admin Pgms (MainMenu, Connection, Requester, Service Name).

In the JCL in MAKLKED (from the PDS "YOURHLQ".BEATCPC.JCL"), complete the following steps.

1. Modify the JOBCARD to your standards.
2. Modify the variables YOURHLQ, TCPLIB and LNKLIB for your PDS.

   Note: LNKLIB is where your CICS region executable binary file resides.
3. The following tasks are specific to the TCP/IP product you are running.
   - For IBM TCP/IP, proceed as follows.

   a. Delete steps 01 and 02.

   b. Delete all lines that read TCPLIB='YOURILINK.T09LOAD'

   - For Interlink TCP/IP, proceed as follows.

   a. Delete steps 1 and 2.

   b. Delete all lines that read TCPLIB='YOUR.SEZATCP'
4. Submit MAKLKED.
5. Verify the results.

Verify the CICS Setup

Log on to your CICS region to install and verify the components you set up for BEA eLink TCP for CICS.

- CEDA INSTALL GROUP(____)

  This is the group name you used in CSDU

- CEMT INQUIRE PROG(____)

  Verify the mapsets, programs, and transids

- CEMT INQUIRE FILE(____)

  Verify the VSAM files

Assemble and Link the CICS DCT

1. Add the following entries to the CICS DCT.

Entry	Link
BEALOG DFHDCT TYPE=SDSCI,	+00450000
DSCNAME=BEALOG,	+00460000
BLKSIZE=136,	+00470000
RECSIZE=132,	+00480000
RECFORM=VARUNB,	+00490000
TYPEFLE=OUTPUT,	+00500000
BUFNO=1	+00510000
BEAL DFHDCT TYPE=EXTRA,	+00690000

Entry	Link
DESTID=BEAL,	+00700000
DSCNAME=BEALOG	+00710000

2. Assemble and link the DCT.

Modify the CICS Startup JCL

1. Add the BEA eLink load library (LNKLIB) to the DFHRPL concatenation.
2. Add the following DD statement to CICS file concatenation.

   ```
   //BEALOG DD SYSOUT=*, DCB=(DSORG=PS, RECFM=V, BLKSIZE=136)
   ```

3. Add DD statement to make the connection between the logical names of the VSAM files and the DSN.

   Note: If DSN statements were added to the CSDU, this step is not necessary.

The following is an example of a CICS startup JCL.

Listing 3-14 Sample JCL to Start eLink TCP for CICS

```
//CICSREG JOB
//CICSCNTL EXEC PGM=IDCAMS,REGION=1M
//SYSPRINT DD SYSOUT=*
//SYSIN DD DISP=SHR,DSN=CICS410.SYSIN(DFHRCYES)
//DTCNTL EXEC PGM=IDCAMS,REGION=1M
//SYSPRINT DD SYSOUT=*
//SYSIN DD DISP=SHR,DSN=CICS410.SYSIN(DFHRCNO)
//*
// CICS EXEC PGM=DFHSIP, REGION=32M, TIME=1439, COND=(1,NE,CICSCNTL),
// =START=COLD,SYSIN,SEC=NO,GRPLIST=ETPLIST
// STEPLIB DD DISP=SHR,DSN=CICS410.SVTSC.SDFHAUTH
// DD DISP=SHR,DSN=CICS410.SDFHAUTH
// DD DISP=SHR,DSN=MASTCAT.MQM112.SCSQAUTH
// DD DISP=SHR,DSN=COB2140.COB2CICS.MODLIB
```

```
// DD DISP=SHR,DSN=C370.SEDCLINK
// DD DISP=SHR,DSN=ADLE370.V1R3M0.SCEECICS
// DD DISP=SHR,DSN=ADLE370.V1R3M0.SCEERUN
// DFHRPL DD DISP=SHR,DSN=CICS410.SVTSC.SDFHLOAD
// DD DISP=SHR,DSN=CICS410.SDFHLOAD
// DD DISP=SHR,DSN=BEA.BEA100.BEALOAD *BEA Connect
// DD DISP=SHR,DSN=TCPIP.V3R1.SEZALINK
// DD DISP=SHR,DSN=TCPIP.V3R1.SEZATCP
// DD DISP=SHR,DSN=MQM112.SCSQCICS
// DD DISP=SHR,DSN=MQM112.SCSQLOAD
// DD DISP=SHR,DSN=COB2140.COB2CICS
// DD DISP=SHR,DSN=MASTCAT.COB2140.COB2LIB
// DD DISP=SHR,DSN=MASTCAT.PLI230.PLILINK
// DD DISP=SHR,DSN=C370.SEDCLINK
// DD DISP=SHR,DSN=ERW121.SERWLOAD
// DD DISP=SHR,DSN=ADLE370.V1R3M0.SCEECICS
// DD DISP=SHR,DSN=ADLE370.V1R3M0.SCEERUN
// SYSIN DD DISP=SHR,DSN=CICS410.CICS.SYSIN(TCICS)
// DFHCSD DD DISP=SHR,DSN=CICS410.DFHCSD
// DFHCMACD DD DISP=SHR,DSN=CICS410.DFHCMACD
// DFHJ01A DD DISP=SHR,DSN=CICS410.CICS.DFHJ01A
// DFHJ01B DD DISP=SHR,DSN=CICS410.CICS.DFHJ01B
// DFHJ01X DD DISP=SHR,DSN=CICS410.CICS.DFHJ01X
// DFHAUXT DD DISP=SHR, DSN=CICS410.CICS.DFHAUXT, DCB=BUFNO=5
// DFHBUXT DD DISP=SHR, DSN=CICS410.CICS.DFHBUXT, DCB=BUFNO=5
// DFHDMPA DD DISP=SHR,DSN=CICS410.CICS.DFHDMPA
// DFHDMPB DD DISP=SHR,DSN=CICS410.CICS.DFHDMPB
// DFHRSD DD DISP=SHR,DSN=CICS410.CICS.DFHRSD
// DFHGCD DD DISP=SHR,DSN=CICS410.CICS.DFHGCD
// DFHLCD DD DISP=SHR,DSN=CICS410.CICS.DFHLCD
// DFHXRMSG DD DISP=SHR, DSN=CICS410.CNTL.CICS.DFHXRMSG
// DFHXRCTL DD DISP=SHR,DSN=CICS410.CNTL.CICS.DFHXRCTL
// DFHTEMP DD DISP=SHR,DSN=CICS410.CNTL.CICS.DFHTEMP
// DFHINTRA DD DISP=SHR,DSN=CICS410.CNTL.CICS.DFHINTRA
// DFHJACD DD DISP=SHR,DSN=CICS410.CICS.DFHJACD
// DFHJPDS DD DISP=SHR,DSN=CICS410.DFHJPDS
// DFHJOUT DD SYSOUT=(A,INTRDR)
// *CEEMSG DD DSN=CEE120.CICSDCT.CEEMSG,DISP=SHR ** AD/CYC LE/370
// *CEEOUT DD DSN=CEE120.CICSDCT.CEEOUT,DISP=SHR ** AD/CYC LE/370
// DFHCXRF DD SYSOUT=X
// MQMMSG DD SYSOUT=*
// LOGUSR DD SYSOUT=*, DCB=(DSORG=PS,RECFM=V, BLKSIZE=136)
// MSGUSR DD SYSOUT=*, DCB=(DSORG=PS,RECFM=V, BLKSIZE=136)
// TCPDATA DD SYSOUT=*, DCB=(DSORG=PS,RECFM=V, BLKSIZE=136)
// BEALOG DD SYSOUT=*, DCB=(DSORG=PS,RECFM=V, BLKSIZE=136) ** BEA Connect
// COUT DD SYSOUT=*, DCB=(DSORG=PS,RECFM=V, BLKSIZE=137)
// PLIMSG DD SYSOUT=*, DCB=(DSORG=PS,RECFM=V, BLKSIZE=137)
// COUT DD SYSOUT=*, DCB=(DSORG=PS,RECFM=V, BLKSIZE=137)
// SYSABEND DD SYSOUT=*
```

```
// TRACEOUT DD SYSOUT=*
// PRINTER DD SYSOUT=*,DCB=BLKSIZE=121
// *
// PRTDMPA EXEC PGM=DFHDUP, PARM=SINGLE, REGION=0M, COND=(1,NE,DTCNTL)
// STEPLIB DD DSN=CICS410.SDFHLOAD,DISP=SHR
// SYSIN DD DUMMY
// DFHDMPDS DD DISP=SHR,DSN=CICS410.CICS.DFHDMPA
// DFHTINDX DD SYSOUT=*
// SYSPRINT DD SYSOUT=*
// DFHPRINT DD SYSOUT=*
// *
// PRTDMPB EXEC PGM=DFHDUP, PARM=SINGLE, REGION=0M, COND=(1,NE,DTCNTL)
// STEPLIB DD DSN=CICS410.SDFHLOAD,DISP=SHR
// SYSIN DD DUMMY
// DFHDMPDS DD DISP=SHR,DSN=CICS410.CICS.DFHDMPB
// DFHTINDX DD SYSOUT=*
// SYSPRINT DD SYSOUT=*
// DFHPRINT DD SYSOUT=*
// *
// * PRTAUXT EXEC PGM=DFHTUP, REGION=0M, COND=(1,NE,DTCNTL)
// * STEPLIB DD DSN=CICS410.SDFHLOAD,DISP=SHR
// * DFHAUXT DD DISP=OLD,DSN=CICS410.CICS.DFHAUXT
// * DFHAXPRT DD SYSOUT=*
// * DFHAXPRM DD DUMMY
// *
// PRTBUXT EXEC PGM=DFHTUP, REGION=0M, COND=(1,NE,DTCNTL)
// STEPLIB DD DSN=CICS410.SDFHLOAD,DISP=SHR
// DFHAUXT DD DISP=SHR,DSN=CICS410.CICS.DFHBUXT
// DFHAXPRT DD SYSOUT=*
// DFHAXPRM DD DUMMY
```

List of Distribution Files and Members

The following tables list and describe the members that make up each distribution partitioned data set.

BEATCPC.DIST.CONTROL

Verify that the following members are installed by BEA eLink TCP for CICS.

Member	Description
CSDU	Del/Def Parameters for (PCTs, PPTs, FCTs)
VCON	Del/Def Parameters for (VSAM Connections)
VREQ	Del/Def Parameters for (VSAM Requesters)
VSVC	Del/Def Parameters for (VSAM Service Names)
VUSR	Del/Def Parameters for (VSAM User Connections)
VISN	Del/Def Parameters for (VSAM Inbound Services)

BEATCPC.DIST.JCL

Verify that the following members are installed by BEA eLink TCP for CICS.

Member	Description
BLDVSAM	Jcl to Del/Def VSAM files (BEAVCON, BEAVREQ, BEAVSVC, BEAVUSR, BEAVISN)
CSDUPD	Jcl to Del/Def CICS (PCTs, PPTs, FCTs)
CMPBC	Proc to Compile/link C program (Configuration)
MAKBC	Jcl to Compile/link C program (Configuration)
CMPLMAP	Proc to Link CICS Mapsets(TCP Admin)
MAKLMAP	Jcl to Link CICS Mapsets(TCP Admin)
MAKL0	Jcl to PreLink/Link Pgm objects(PreRequester, Shutdown)
MAKL1	Jcl to PreLink/Link Pgm objects(TCP Admin)

Member	Description
MAKL3	Jcl to PreLink/Link Pgm objects(Requester, Handler) -IBM SOCKET-
MAKL2	Jcl to PreLink/Link Pgm objects(Requester, Handler) -INTERLINK-
CMPLKED	Proc to Link the PreLinked PLKED
MAKLKED	Jcl to Link the PreLinked PLKED
LNKIBM	Proc to PreLink/Link Pgm objects
LNKINT	Proc to PreLink/Link Pgm objects(Requester, Handler) -INTERLINK-

BEATCPC.DIST.SOURCE

Verify that the following members are installed by BEA eLink TCP for CICS.

Member	Description
BEACFGSV	Program Name Configuration (Files, Maps, Pgms, Trns)
BEACCLN1	Program Sample Test Client
BEACSVR1	Program Sample Test Server
T09CONFG	Macro of BEAH Transid for -INTERLINK-

BEATCPC.DIST.INCLUDE

Verify that the following members are installed by BEA eLink TCP for CICS.

Member	Description
CLIENT	Copybook for COBOL CICS Client Request/Response Header

Member	Description
CLIENTH	Include for C CICS Client Request/Response Header
TWACOPY	Copybook for COBOL CICS Server TWA Message size
TWAINCL	Include for C CICS Server TWA Message size
BEACFGSV	Include for the name configuration (Files, Maps, Pgms, Trns)

BEATCPC.DIST.PLKED

Verify that the following members are installed by BEA eLink TCP for CICS.

Member	Description
BEACIC0I	Prelinked object of Handler -INTERLINK-
BEACIC00	Prelinked object of Handler -IBM SOCKET-
BEAPCON	Prelinked object of Admin Connections
BEAPMNU	Prelinked object of Admin(Main Menu)
BEAPREQ	Prelinked object of Admin(Requesters)
BEAPRERQ	Prelinked object of Pre-Requester
BEAPSVC	Prelinked object of Admin(Outbound Service Names)
BEAREQST	Prelinked object of Requester -IBM SOCKET-
BEAREQSI	Prelinked object of Requester -INTERLINK-
BEASHUT	Prelinked object of Shutdown
BEAAPPLC	Prelinked object of Application Handler
BEAPUSR	Prelinked object of Admin(User Connection Account)
BEAPISN	Prelinked object of Admin(Inbound Service Names)
CNVTISNC	Prelinked object of Configuration Converter

BEATCPC.DIST.OBJECT

Verify that the following members are installed by BEA eLink TCP for CICS:

- BEAAPPLC
- BEACFGSV
- BEACIC00
- BEAERRNO
- BEAEVT2
- BEALOGC
- BEAMCON
- BEAMMNU
- BEAMREQ
- BEAMSUB
- BEAMSVC
- BEAMUSR
- BEAMISN
- BEAPCON
- BEAPMNU
- BEAPREQ
- BEAPRERQ
- BEAPSVC
- BEAPUSR
- BEAPISN
- BEAREQST
- BEARQSUB

- BEASHUT
- EBCASC
- BEASUBS
- BEAGHBN
- BEASKIBM
- BEASKINT
- BEASKVAR
- CNVTISNV

A Changing Resource Names

The default names for BEA eLink TCP for CICS Mapsets, Programs, Transids, and VSAMs are listed in the following sections. If you want to use a different naming convention, use the procedure for changing names included after each table.

Note: You must have the C 370 compiler to customize resource names.

Contents of SOURCE(BEACFGSV)

The names in the following table are located in PDS "YOURHLQ".BEATCPC.SOURCE(BEACFGSV)

Name	Type	Description
BEAM	Transid	Main Menu
BEAPMNU	Program	Main Menu
BEAMMNU	Mapset	Main Menu
BEAC	Transid	Connection Administration
BEAPCON	Program	Connection Administration
BEAMCON	Mapset	Connection Administration

Name	Type	Description
BEAR	Transid	Requester Administration
BEAPREQ	Program	Requester Administration
BEAMREQ	Mapset	Requester Administration
BEAS	Transid	Service Name Administration
BEAPSVC	Program	Service Name Administration
BEAMSVC	Mapset	Service Name Administration
BEAU	Transid	User Connection Administration
BEAPUSR	Program	User Connection Administration
BEAMUSR	Mapset	User Connection Administration
BEAI	Transid	Inbound Service Administration
BEAPISN	Program	Inbound Service Administration
BEAMISN	Mapset	Inbound Service Administration
BEAVCON	VSAM	Connection Table
BEAVREQ	VSAM	Requester Table
BEAVSVC	VSAM	Service Name Table
BEAVUSR	VSAM	User Connection Table
BEAVISN	VSAM	Inbound Service Table
BEAPRERQ	Program	Pre-requester
BREQ	Transid	To start Requester
BEAL	TDQid	Log Message TD Queue
BEAA	Transid	Application Handler Transid
BEAHBDWN	TSQid	Handler Shutdown TS Queue

Changing Resource Names for SOURCE(BEACFGSV)

To change any of the default names shown in the previous table, complete the following steps.

1. Modify Program (BEACFGSV) located in PDS "YOURHLQ".BEATCPC.SOURCE
2. For the JCL (MAKBC) located in PDS "YOURHLQ".BEATCPC.JCL, complete the following steps.
   a. Modify the JOBCARD to your standards.
   b. Modify "YOURHLQ.BEATCPC.SOURCE" and "YOURHLQ.BEATCPC.OBJECT".
   c. Submit MAKBC.
3. Verify the results.

Contents of CONTROL(CSDU)

The names in the following table are located in PDS "YOURHLQ".BEATCPC.CONTROL(CSDU)

Name	Type	Description
BEAPCON	Program	Connection administration
BEAPMNU	Program	Main menu
BEAPREQ	Program	Requester administration
BEAPSVC	Program	Outbound service administration
BEAPUSR	Program	User connection administration
BEAPISN	Program	Inbound service administration

Name	Type	Description
BEASHUT	Program	Shutdown
BEAPRERQ	Program	Pre-Requester
BEAREQST	Program	Requester
BEACIC00	Program	Handler
BEAAPPLC	Program	Application Handler
BEAC	Transaction	Connection administration
BEAM	Transaction	Main menu
BEAR	Transaction	Requester administration
BEAS	Transaction	Outbound service administration
BEAU	Transaction	User connection administration
BEAI	Transaction	Inbound service administration
BDWN	Transaction	Shutdown
BREQ	Transaction	Requester
BEAH	Transaction	Handler
BEAA	Transaction	Application Handler
BEAMCON	Mapset	Connection administration
BEAMMNU	Mapset	Main menu
BEAMREQ	Mapset	Requester administration
BEAMSVC	Mapset	Outbound service administration
BEAMUSR	Mapset	User connection administration
BEAMISN	Mapset	Inbound service administration
BEAVCON	File	Connection table
BEAVREQ	File	Requester table

Name	Type	Description
BEAVSVC	File	Outbound service table
BEAVUSR	File	User connection table
BEAVISN	File	Inbound service table

Changing Resource Names for CONTROL(CSDU)

After changing resource names in BEACFGSV, you need to modify the file (CSDU) located in PDS "YOURHLQ".BEATCPC.CONTROL.

Changes to the CSDU member must correspond to changes in the BEACFGSV member.

You can also change the names of the BDWN and BEAH transactions. If the name of the BEAH transaction changes, the CICSHAND parameter in the GWICONFIG file of the BEA eLink TCP for TUXEDO gateway must also reflect the change.

Index

D

E

F

G

H

I

U

V

W