

# BS2000/OSD Environments

This section provides information about installing and running Entire Net-Work in the BS2000/OSD operating system environment. It covers the following topics:

- Installation Tape Information
- Non-SMA Installation Procedure
- SMA Installation Instructions
- Running in BS2000/OSD Environments
- Starting Entire Net-Work
- Preparing and Installing the Product License File
- Output File Concept
- Entering Operator Commands
- NETEND Utility

The information in this section uses the substitution variable *vrs*, which stands for the current version, revision, and system maintenance (SM) level of the product; for example: "WCP621".

Refer to the Entire Net-Work *Report of Tape Creation* for the names and sequence of the libraries and files on the installation tape.

---

## Installation Tape Information

### Tape Contents

The Entire Net-Work installation tape for BS2000/OSD contains all components required to run Entire Net-Work as sequential files:

- Entire Net-Work Mainline Processor and Service Routines
- Entire Net-Work TCP/IP Line Driver (if ordered)
- Adabas components required to run Entire Net-Work if Adabas is not installed
- Installation-dependent files

### Disk Contents

After successful installation, the disk will contain the following LMS libraries:

Library Name	Description
WCPvrs.LIB	Entire Net-Work Main Line and Service Routines
WTCvrs.LIB	Entire Net-Work TCP/IP Line Driver (optional)

All load modules, sources, and jobs are contained in the appropriate libraries.

## Prerequisites

Verify that any required updates are applied to the WAL 8 libraries so it runs Entire Net-Work Administration in BS2000 environments.

## Non-SMA Installation Procedure

If you are not using Software AG's System Maintenance Aid (SMA), follow the instructions in this section.

- Step 1: Copy the Contents of the Tape to Disk
- Step 2: Apply Corrective Maintenance
- Step 3: Perform the Line Driver-Dependent Installations

### Step 1: Copy the Contents of the Tape to Disk

#### Note:

If you are using SMA, refer to the *System Maintenance Aid* documentation. If you are not using SMA, follow the instructions below.

This section explains how to copy all files from tape to disk. You will then need to perform the individual installation procedure for each component to be installed.

- Step 1: Copy the Library SRVvrs.LIB from Tape to Disk
- Step 2: Copy the Procedure COPY.PROC from Tape to Disk
- Step 3: Copy all Product Files from Tape to Disk

#### Step 1: Copy the Library SRVvrs.LIB from Tape to Disk

This step is not necessary if you have already copied the library SRVvrs.LIB from another Software AG installation tape. For further information, refer to the element #READ-ME in this library. The library SRVvrs.LIB is stored on the tape as a sequential file named SRVvrs.LIBS containing LMS commands. The current version vrs can be obtained from the *Report of Tape Creation*. To convert this sequential file into an LMS library, execute the following commands:

```
/IMPORT-FILE SUPPORT=*TAPE(FILE-NAME=SRVvrs.LIBS,-
/ VOLUME=volser, DEV-TYPE=tape-device)
/ADD-FILE-LINK LINK-NAME=EDTSAM, FILE-NAME=SRVvrs.LIBS,-
/ SUPPORT=*TAPE(FILE-SEQ=3), ACC-METH=*BY-CAT,-
/ BUF-LEN=*BY-CAT, REC-FORM=*BY-CAT, REC-SIZE=*BY-CAT
```

```

/START-EDT
@READ  '/'
@SYSTEM 'REMOVE-FILE-LINK  EDTSAM'
@SYSTEM 'EXPORT-FILE  FILE-NAME=SRVvrs.LIBS'
@WRITE  'SRVvrs.LIBS'
@HALT
/ASS-SYSDTA  SRVvrs.LIBS
/MOD-JOB-SW  ON=1
/START-PROG  $LMS
/MOD-JOB-SW  OFF=1
/ASS-SYSDTA  *PRIMARY

```

where:

*tape-device* is the device type of the tape, for example: TAPE-C4  
*volser* is the VOLSER of the tape (see *Report of Tape Creation*)

## Step 2: Copy the Procedure COPY.PROC from Tape to Disk

To copy the procedure COPY . PROC to disk, call the procedure P . COPYTAPE in the library SRVvrs . LIB:

```

/CALL-PROCEDURE  (SRVvrs.LIB,P.COPYTAPE), -
/  (VSNT=volser, DEVT=tape-device)

```

If you use a TAPE-C4 device, you may omit the parameter DEVT.

## Step 3: Copy all Product Files from Tape to Disk

To copy all Software AG product files from tape to disk, enter the procedure COPY . PROC:

```

/ENTER-PROCEDURE  COPY.PROC, DEVT=tape-device

```

If you use a TAPE-C4 device, you may omit the parameter DEVT. The result of this procedure is written to the file L . REPORT . SRV.

## Step 2: Apply Corrective Maintenance

Apply corrective maintenance if necessary (refer to the *Report of Tape Creation*).



### Warning:

**Zap WM621028 must be applied in Entire Net-Work 6.2 SP1 installations. This zap must be applied to all nodes that will connect together, regardless of the operating system or whether Adabas Cluster Services exists on the node. If an Entire Net-Work 6.2 SP1 node without this zap attempts to connect to a node that has it applied, message NETT029I will be issued and the connection will fail. If a node without zap WM621028 attempts to connect via IUCV to a node that has the zap applied, one of the nodes will get message NET0151I and the connection will fail.**

### Step 3: Perform the Line Driver-Dependent Installations

Perform the line driver-dependent installation procedure and prepare the DRIVER and LINK statements for the TCPX line driver.

## SMA Installation Instructions

If you are using Software AG's System Maintenance Aid (SMA) install, proceed as usual. After installation, the PLAM libraries WCPvrs.LIB and optionally WTCvrs.LIB will exist on your disk and be ready for use.

For more information about the SMA installation process, refer to the System Maintenance Aid documentation.

## Running in BS2000/OSD Environments

The following points should be considered when running Entire Net-Work in the BS2000/OSD environment:

- Correctly configured TCP/IP environments are prerequisites for starting Entire Net-Work.
- The library WCPvrs.LIB contains a set of predefined elements that is usually ready to use. The J-Element START can be used to start Entire Net-Work with a minimum set of configuration data. It contains an example configuration including a DCAM and a TCP/IP connection.
- If the new tracing facilities are required, set the TRACE parameter to YES in the call to the STARTP procedure.
- In order to define which Adabas modules will be used to run Entire Net-Work, either:
  1. Create a JOB-VARIABLE named 'ADALIB' and set it to the correct name of the Adabas or WAL library, or
  2. Modify the START procedure so that the name is supplied as the ADALIB parameter in the call to STARTP, as follows:

```

/CALL-PROC -
/ FROM-FILE = *LIB(LIB = WCPvrs.LIB, -
/           ELE = STARTP), -
/ PROC-PAR = (DUMP = YES, -
/           ADALIB= xxxxxx, -
/           TRACE= NO, -
/           OPMSG = YES, -
/           TIME = NO), -
/ LOGGING = *YES

```

where xxxxxx is the name of your library. For example: \$SAG.WAL814.MOD

- If the Simple Connection Line Driver is to be used, then an extra task will be started automatically. The parameters for starting this task are defined in the variable NWBS2SUB within the STARTP procedure. The library or procedure names may be altered if required (see the section *Starting the TCP/IP Line Driver*).

# Starting Entire Net-Work

To actually start Entire Net-Work, enter the following command:

```
/ENTER-JOB (WCPvrs.LIB,START)
```

The SAG supplied J-Element START is executed. It has the following sample content:

```
/SET-LOGON-PARAMETERS -
/  USER-ID   = *NO,-
/  ACCOUNT   = *NONE,-
/  JOB-CLASS = *STD,-
/  JOB-NAME  = NETWORK
/ASSIGN-SYSOUT -
/  TO       = START.JNL
/CALL-PROC -
/  FROM-FILE = *LIB(LIB = WCPvrs.LIB,-
/              ELE = STARTP),-
/  PROC-PAR  = (DUMP = YES,-
/              ADALIB= &(ADALIB),-
/              LICLIB= LICvrs.MOD,-
/              WCPLIB = WCPvrs.LIB,-
/              WTCLIB = WTCvrs.LIB,-
/              MLCLIB = MLCmvrs.LIB,-
/              VERSION = vrs,-
/              TRACE = NO,-
/              OPMSG = YES,-
/              TIME  = NO),-
/  LOGGING   = *YES
ADARUN PROG=NETWRK,TARGETID=vrs2,IDTNAME=ADANET02
ADARUN TASKCTGY=TP,FORCE=YES,GROUPS=NO,CMDQMODE=ABOVE
* The next 2 lines are only needed when running with a
* Sockets version less than 2.2
ADARUN CMSIZE=32000000,CMFIX=Y,CMSCOPE=GROUP,CMADDR=X'01300000'
ADARUN CMLSIZE=4096000,CMLFIX=Y,CMLSCOPE=GROUP
/EOF
NODE      MHK2 NT=128,LOG=N,PASSWORD=ALL,-
          TRANS DEF ID=(0,0),TOASCII=E2ASIE,TOEBCDIC=A2ESIE
DRIVER    DCAM
DRIVER    SMH  API=BS2,SERVERID=8010
DRIVER    TCPI API=BS2,ACCEPTUI=N,SERVERID=8882,APITRACE=N,-
          OPTIONS1=(0,0,0,0,0,0,1024)
LINK      MHK1 DCAM PRONAME=HOST
LINK      SMHLINK SMH
LINK      IBM TCPI I=157.189.160.95,SERVERID=2000
LINK      H60 TCPI ADJHOST=SIH1,SERVERID=8881
LINK      PCMHK TCPI ADJHOST=PCMHK,SERVERID=8882
/EXIT-JOB -
/  MODE     = NORMAL,-
/  SYS-OUT  = NONE
```

## Checking the Sockets Version

To determine the version of Sockets that is running, input the following command:

```
SHOW-SUBSYSTEM-STATUS SUBSYSTEM-NAME=SOCKETS
```

The Sockets version number will be returned.

## JCL Notes for Running Sockets

The default CMSIZE is around 32MB. We recommend that this size be set to the largest possible in the address space. This area is used by all drivers by the buffer pool manager. It is important to set this as large as possible, particularly when transferring large data objects. This system runs a subtask called NETTCP.

If optional zap WY622002 is applied, the ADARUN parameters CMSIZE, CMFIX, CMSCOPE, CMADDR, CMLSIZE, CMLFIX and CMLSCOPE need not be applied. Here, the ability to pass large objects through Entire Net-Work is dependent on the size of the address space alone. This system does not run a subtask NETTCP.

### Important:

Do not change the parameters CMFIX=Y, CMSCOPE=GROUP, or CMADDR=X'01300000'

The extra ADARUN entries, beginning with CMSIZE and CMLSIZE, are required for the Simple Connection Line Driver.

If you wish to use a different job class for Entire Net-Work, it may be supplied to the START procedure as follows (xxxxxxx is your defined JOB-CLASS):

```
/SET-LOGON-PARAMETERS -
/  USER-ID    = *NO, -
/  ACCOUNT    = *NONE, -
/  JOB-CLASS  = xxxxxxxx, -
/  JOB-NAME   = NETWORK
```

START calls procedure STARTP, which is shown below:

```
/* ***** */
/*                                     */
/*           Entire Net-Work STARTP Procedure           */
/*                                     */
/*  MODE SINGLE: ONLY ONE NET-WORK SESSION PER USER-ID ALLOWED  */
/*                                     */
/* ***** */
/SET-PROCEDURE-OPTIONS -
/  IMPLICIT-DECLARATION = *YES, -          &* For SDF-BASYS
/  LOGGING-ALLOWED     = *YES, -          &*
/  SYSTEM-FIL-CONTEXT  = *SAME-AS-CAL, - &* For ADARUN
/  JV-REPLACEMENT      = *AFTER, -       &* Allow JV for ADALIB
/  ERROR-MECHANISM     = *SPIN-OFF      &* For SDF-BASYS
/BEGIN-PARAMETER-DECLARATION
/  DECLARE-PARAM -                          &* Used to form WCP and WTC
/    VERSION(INIT='621')                    &* library names.
/  DECLARE-PARAM -                          &* Adabas or WAL library.
/    ADALIB                                  &*
/  DECLARE-PARAM -                          &* License Module
/    LICLIB                                  &* Library
/  DECLARE-PARAM -                          &*
/    WCPLIB                                  &* Net-Work main library
/  DECLARE-PARAM -                          &*
/    WTCLIB                                  &* Net-Work TCP/IP Library
/  DECLARE-PARAM -                          &*
```

```

/      MLCLIB                                &* License Library
/  DECLARE-PARAM -                          &* Determines whether a dump
/      DUMP (INIT='NO')                    &* is created for errors.
/  DECLARE-PARAM -                          &* For tracing facilities
/      TRACE-TO-NETPRNT (INIT='NO')       &*
/  DECLARE-PARAM -                          &* TCP/IP operator messages.
/      OPMSG (INIT='YES')                 &*
/  DECLARE-PARAM -                          &* CPU limit for TCP sub task;
/      TIME (INIT='NO')                   &* NO, STD or 1 - 32767
/END-PARAMETER-DECLARATION
/
/  IF (OPMSG = 'NO')
/      MOD-USER-SW ON = 1
/  END-IF
/
/  MODIFY-JOB-OPTIONS -                    &* Log messages to the system
/      OPERATOR = *YES                    &* Console.
/
/  MODIFY-TEST-OPTIONS -
/      DUMP      = &DUMP
/
/  CALL-PROC -
/      FROM-FILE = *LIB(LIB = &(WCPLIB), -
/                      ELE = NEWGEN), -
/      PROC-PAR  = (NET.PROTO.OUT)        &* Create the next generation
/                                          &* in this file group, then
/  ASSIGN-SYSOUT -                        &* direct SYSOUT to it.
/      TO        = NET.PROTO.OUT(+0)
/
/  IF (TRACE-TO-NETPRNT = 'YES')
/      CALL-PROC
/          FROM-FILE = *LIB(LIB = &(WCPLIB), -
/                          ELE = NEWGEN), -
/          PROC-PAR  = (NET.NETPRNT)      &* Create the next generation
/                                          &* in this file group, then
/  SET-FILE-LINK -                        &* direct NETPRNT tracing
/          LINK-NAME = NETPRNT,-          &* to it.
/          FILE-NAME = NET.NETPRNT(+0), -
/          ACC-METH  = SAM,-
/          REC-FORM  = *FIXED(REC-SIZ=121)
/
/END-IF
/  SHOW-JOB-STATUS
/
/  *-----*
/  "      The following section is for the TCP/IP driver      "
/  "
/  " 1) The task is started according to the parameters supplied in the "
/  "    variable NWBS2SUB. "
/  " 2) If a different procedure is to be used, the LIB and ELE names "
/  "    in the FROM parameter may be altered. "
/  " 3) If the subtask code is to be loaded from a different library, "
/  "    then the WTCLIB name in the PROC-PAR parameter may be changed. "
/  "
/  "      **** Do NOT start the subtask manually! ****      "
/  *-----*
/  SHOW-FILE-ATTRIBUTES -                  &* Test whether TCP/IP driver
/      FILE-NAME = &(WTCLIB),-            &* is available. If it is...
/      INFORMAT  = *MIN
/
/  SET-FILE-LINK -                          &* add WTC file link for BLS,
/      LINK-NAME = BLSLIB01, -

```

```

/          FILE-NAME = &(WTCLIB)
/
/ DECLARE-VARIABLE -          &* then set up the parameters
/          NWBS2SUB(INIT=-    &* for the NWBS2SUB task.
/          /'ENTER-PROC FROM  = (LIB = WCP&VERSION.LIB,-
/          /          ELE = SUBTASKPROC),-
/          /          PROC-PAR= (WTCLIB= &(WCPLIB),-
/          /          WCPLIB= &(WTCLIB),-
/          /          DUMP = &DUMP),-
/          /          JOB-CLAS= *STD,-
/          /          JOB-NAME= NETTCP,-
/          /          RESOURCE= *PAR(CPU-LIMIT=&TIME),-
/          /          SYS-OUTP=
*DELETE')

/SET-JOB-STEP          &* Catch spinoff
/
/ SET-FILE-LINK -          &* add WCP file link for BLS.
/          LINK-NAME = BLSLIB00, -
/          FILE-NAME = &(WCPLIB)
/
/ SET-FILE-LINK -          &* add a file link for ADARUN.
/          LINK-NAME = DDLIB, -
/          FILE-NAME = &ADALIB
/ SET-FILE-LINK -          &* Add a file link for BLS.
/          LINK-NAME = BLSLIB02, -
/          FILE-NAME = &(ADALIB)
/
/ SET-FILE-LINK -          &* Add a file link for BLS.
/          LINK-NAME = BLSLIB03, -
/          FILE-NAME = &(LICLIB)
/
/ SET-FILE-LINK -          &* Add a file link for BLS.
/          LINK-NAME = BLSLIB04, -
/          FILE-NAME = &(MLCLIB)
/
/
/ SEND-MSG -
/          TO = *OPERATOR,-
/          MSG = 'STARTING Entire Net-Work (V &VERSION)
/
/ START-PROGRAM -
/          FROM          = *MODULE(LIB = &ADALIB, ELEM = ADARUN, PROG = ANY,-
/          /          RUN-MOD = ADV(ALT-LIB = YES))
/
/ IF (OPERATOR-MSG = 'NO')          &* Unset switch 1 if
/          MOD-USER-SW OFF = 1          &* it was set earlier.
/ END-IF          &*
/
/EXIT-PROC

```

**Notes:**

1. Modify OPMSG=NO if possible. This disables all dependent Simple Connection Line Driver operator messages.
2. Modify DUMP=YES if possible. This allows your software AG representative to assist you more quickly, should this be necessary.
3. As Entire Net-Work starts, it will display a number of messages on the system console. When it has initialized successfully, it will display the message: NET1000I: N E T - W O R K node MHK2 active
4. This messages is preceded by several other messages that are also displayed on the system console.



## Preparing and Installing the Product License File

You must install a valid license on all mainframe platforms in which your Software AG mainframe product is installed. The license file is provided as an XML document. The document's encoding is US-ASCII and must remain in that format -- even on the mainframe. It must not be modified. Any modification of the license file will invalidate the digital signature and the license check will fail. If the check fails, you will not be able to run the product. In the event of a check failure, contact your Software AG technical support representative for assistance.

### Note:

Thirty days before the license expires, license check failure messages are produced. Your software product will still function, but these messages warn you that it is time to obtain a new license.

In this step, you will prepare the license file (obtain it from e-mail or the installation tape and store it on your z/OS system) and then install it:

- Preparing the Product License File
- Installing the Product License File

## Preparing the Product License File

The product license file is supplied on the individual customer installation tape or separately via an e-mail attachment. Before you can install the license, you must transfer it from e-mail or the installation tape and store it on a z/OS system. This section describes how to do this for a license distributed using either method.

### ▶ To prepare the license file from an e-mail attachment, complete the following steps:

1. Transfer the license to BS2000, as described in *Transferring a License File from PC to a BS2000 Host Using FTP*.
2. Verify that the transferred license file is a SAM sequential file, taking care to preserve its format as ASCII.

### ▶ To prepare the license file from the installation tape, complete the following steps:

- Verify that the transferred license file is a SAM sequential file, taking care to preserve its format as ASCII.

Check the license file by modifying and running the job `WCPvrs.LIB(LICCHECK,J)`.

## Installing the Product License File

### ▶ To install the product license file, complete the following steps:

1. Convert the license to a loadable element (NETLIC) that is then loaded by Entire Net-Work by modifying and running the job `WCPvrs.LIB(NETLICA,J)`.

This will put the NETLIC into a license module library.

2. Update your Entire Net-Work startup jobs to reference the license library. Complete the following steps:
  1. Make sure any previously-created NETLIC module is inaccessible to the Entire Net-Work load library being used in your Entire Net-Work startup job. Entire Net-Work first tries to load NETLIC, and, if unsuccessful, then reads from a DDLIC link name referenced in the Entire Net-Work startup job.
  2. Software AG licensing requires that the modules LICMAIN(R) and LICUTIL(R) be loaded when Entire Net-Work starts up. These modules are distributed in the MLCvrs.MOD library. Do either (but not both) of the following:
    - Copy LICMAIN and LICUTIL into the WCPvrs.LIB library.
    - Concatenate MLCvrs.MOD into the BLSLIB chain with WCPvrs.LIB.
  3. Verify that the license file is correctly referenced in each startup job. Do either (but not both) of the following:
    - Verify that the NETLIC load module is stored in a load library that is accessible to the Entire Net-Work load library. Add the user load library in which NETLIC resides to the BLSLIB concatenation of the Entire Net-Work startup job or copy the NETLIC library into the WCPvrs.LIB library.
    - Verify that there is *no* NETLIC load module accessible to the Entire Net-Work load library and that the following link statement is included in the WCPvrs.LIB job STARTP (J):

```
/ADD-FILE-LINK DDLIC,license_file
```

where *license\_file* is the name of the license file loaded from the tape (in ASCII format). Entire Net-Work first tries to load NETLIC and, if unsuccessful, it reads from the DDLIC link name.

**Note:**

To read the license file via the DDLIC link name, the correction WCP621L001 (library WCP621.BS2L001) must be applied.

## Output File Concept

Entire Net-Work produces 2 FGGs with 7 generations, which are:

FGG	Description
NET.PROTO.OUT	contains all main task output*
NET.PROTO.TCP.OUT (Sockets 2.0 only)	contains all subtask output (TCP/IP)*
NET.NETPRNT	(optional) contains tracing data

\*All console messages are also contained in the above output files.

## Entering Operator Commands

Entire Net-Work operator commands (as described in their corresponding sections) are generally issued as follows:

```
/INF-PROG 'message text',*TSN(<tsn>)
```

```
/INTR <tsn>,<command>
```

<tsn> is the task sequence number of Entire Net-Work's main task or subtask

<command> is the command of Entire Net-Work's main task or subtask

The following examples assume the task serial number of the Entire Net-Work task is 1234:

```
/INF-PROG D 'LINKS',*TSN(1234)
```

displays all links currently known by Entire Net-Work.

```
/INF-PROG C 'TOUNIX1',*TSN(1234)
```

connects to foreign node known by link named "tounix1".

```
/INF-PROG C 'MYPC',*TSN(1234)
```

connects to link named "mypc".

```
/INF-PROG 'NETEND',*TSN(1234)
```

shuts down Entire Net-Work.

## NETEND Utility

It is not normally possible to shut down an Entire Net-Work node without access to either a console or a privileged user (\$TSOS).

Entire Net-Work Version 6 provides the NETEND utility procedure, which allows local or remote Entire Net-Work nodes to be shut down without such access. The NETEND procedure can be invoked as follows:

```
CALL-PROC *LIB($SAG.WCPvrs.LIB,NETEND),(  
    ADALIB=<Adabas or WAL library>,  
    IDTNAME= <idtname>,  
    NETLIB=$SAG.WCPvrs.LIB,  
    TARGETID= <node number>)
```

NETEND uses the Programmable Command Interface (PCI). If you wish to use this utility, see the NODE statement keywords PASSWORD, ENDCMS, and REMCMD in the section *Programmable Command Interface*.