

Overview - Shipping Your Application

Introduction.

The purpose of this document is to summarize the information in various locations of our documentation on the topic of “Shipping your Application” and “How to setup Application Server”.

The document is based on information that can be found in:

- Application Server for Net Express v3.1 SP1 installation Help file, Readas31.hlp
- Application Server Licensing User Guide
- Net Express v3.1 SP1 Online Help

This document is based on the information that was available as of July 5th, 2002.

This document does **NOT** replace the product documentation. For up-to-date documentation please refer to the above mentioned sources.

When you ship an application, you need to ship Net Express run-time system and support files with it, and you must ensure that you are licensed to ship those files.

All the Net Express run-time system and support files are supplied as Net Express Application Server. These files are on the Net Express CD, in the directory Application Server, and also on the separate CD, Application Server for Net Express. End-users require licenses to run applications that use Application Server.

See **1.0 - Licensing Application Server** section.

In previous versions, you could pick and choose which Application Server files to ship with your application. We now recommend shipping the whole of Application Server to avoid compatibility issues and other problems. However, you can still ship selected the files if you wish.

See **2.0 - Determining the files to ship** section.

You also need to provide an installation and setup procedure not only for your application but also for Application Server. First you need to decide how your end-users will install your application. For example, will they install your application together with Application Server or will they install a shared Application Server separate from your application?

See **3.0 - Setting up your application** section.

1.0 - Licensing Application Server

You need an Application Server license (AS license) to run an application developed using Net Express. There are broadly two types of AS licenses: developer licenses and end-user licenses.

Developer Licenses

Net Express is shipped with five AS licenses for each developer. They enable you, the developer, to test applications on a separate machine from your development system. For full information, see the chapter Developer Licenses in your *Application Server Licensing Guide*

End-user Licenses

When you purchase Application Server for Net Express for your end-users, you receive a License Key Card. This card is your paper copy of an AS license. It identifies the product name (Application Server for Net Express) and the number of licensed users, and it has printed on it a license key.

The AS licenses are held in a license database. When the end-user runs an application, Application Server checks that a license is available in the license database.

This means that the AS license must be installed before the end-user can run your application. We recommend that you arrange to install the license automatically as part of the installation of your application and Application Server. You can do this using the utility `aslmepsilent.exe`.

See section *C3 - How to install an AS license silently*

If you are not installing the licenses silently, you will need to provide your end-users with the license key. The end-users might also need a record of the license key when solving some problems, such as the license database being moved or corrupted in some way.

End-users require information on licensing and how to run an application that uses Application Server. This information is provided in the Application Server User's Guide. This User's Guide is similar to the Application Server Licensing Guide and indeed some chapters are identical. The two books are:

User's Guide. This is for end-users. It is in the Application Server directory on your Net Express CD and also on the Application Server for Net Express CD. We recommend that you ship this book with your application.

Application Server Licensing Guide. This is for you, the developer. It is installed with Net Express and is on the Net Express bookshelf.

2.0 - Determining the Files to Ship

The Application Server files you need to ship with your application depend on how you linked the application, the type of executable it is, and the support functionality it uses. The files that you need to ship also depend on whether your application is single or multi-threaded.

When you ship a modular application (one that is dynamically linked), you need to ship the following files:

- The main program of your application.
- The Microsoft Visual C++ run-time system, msvcr7.dll.
- Any dynamically linked programs that are part of your application, including objects that have been linked into dynamic link libraries.
- The file for the shared run-time system: cblrtss.dll for single-threaded applications, or cblrtsm.dll for multi-threaded applications.
- The Application Server licensing files. See **A1 - Application Server Licensing Files to Ship** section.
- Any dynamically linked run-time support modules used by your application. See **A2 - Run-time support files to ship** section
- If your application consists of .int files and/or .gnt files, you need to ship some additional support files, and you might need to ship a trigger program. If you use one of the supplied trigger programs, run*.exe, we strongly recommend installing the full Application Server run-time system using the supplied setup, and not shipping individual run-time files as described here. See **A3 - .int and .gnt Support Files to Ship** section
- OO support files, if your application uses OO COBOL. See **A4 - OO COBOL Support Files to Ship** section.
- The on-site debugging tools if necessary. See **A5 - On-site Debugging Support Files to Ship** section
- If your application uses the enhanced ACCEPT/DISPLAY syntax, and you have configured it to use a non-default configuration, you need to ship the file, adisctrl.

You can package the files you need however you choose, such as combining the necessary .gnt files into an .lbr file, or creating .dll files from the supplied .obj files, instead of shipping the .gnt files.

3.0 - Setting Up Application Server

When you ship an application, you need to provide an installation and setup procedure not only for your application but also for Application Server. You also need to set up a license database, which might be on the end-user's machine or on a network server.

The main ways of packaging an application and Application Server are:

A standalone application

A standalone application has Application Server built in, except the licensing files are separate. To ship a standalone application, only the following files are required: the application executable, the Microsoft Visual C++ run-time system, msvcrt.dll and some Application Server licensing files. See **A1 - Application Server licensing files to ship** section.

As part of your application setup, you need to set up the license database and optionally install the AS license silently. See **B1 - How to set up a standalone application** section.

An application with its own Application Server

Each application can have its own copy of Application Server. The application and its Application Server are installed together in the same directory. You can install selected Application Server files, although we recommend installing the full Application Server.

You need to create an installation procedure or setup script that installs the Application Server files. This script must also set up a license database if this is the first time Application Server is being installed on this machine. Subsequent installations use the same license database, so that the database holds a pool of AS licenses for all the copies of Application Server installed on the machine. Finally the script needs to enable an organization to set up a shared license database on a network server. This then holds all the licenses for all the end-users' applications. See **B2 - How to set up an application with its own Application Server** section.

A shared Application Server

A shared Application Server is used by multiple COBOL applications. It is installed in its own directory, separate from any applications. There is only one Application Server installed on a machine.

You use the supplied setup program to install a shared Application Server and set up the relevant registry entries. You can run this setup silently so that end-users are not prompted for information. This setup also enables an organization to set up a shared license database on a network server. This database then holds all the AS licenses for all the end-users' applications.

If your application uses one of the supplied trigger programs, run*.exe, we strongly recommend installing a shared Application Server. See **B3 - How to set up an application with a shared Application Server** section.

A1 - Application Server Licensing Files to Ship

You must ship some of the licensing files, such as the License Administration utility, AppTrack. You can optionally ship other files, such as the setup program for Application Server. The files to ship are:

Mandatory Files

Functionality:	Files you must ship:
AppTrack utility	aptrack.exe
Licensing support files	prodfile
	semfile
National Language Support message files	mflangdf.lbr (default language)
	mflang01.lbr (American English)
	mflang05.lbr (English)
	mflang20.lbr (Japanese)

Optional Files

If you require support for:	Files needed:
Utility to enter license keys silently	aslmepsilent.exe
Utility to set up a license database	aslmeplocate.exe

Application Server setup program	setup.exe
	setup.ini
	setup.ins
	setup.iss
	setup.sss
	setup.bmp
	inst32i.ex
	_isdel.exe
	_setup.dll
	_setup.lib

Initialization file to suppress the prompt for a license key during the setup program supplied by Micro Focus MicroFocusAppServ31.ini

Application Server User's Guide all the files in the subdirectory application server\docs.

All these files are available from the Application Server directory of your Net Express installation or the Application Server for Net Express CD.

A2 - Run-time Support Files to Ship

Many of the frequently required files are already packaged together into the `utils.lbr` file, which you can ship if it suits your application's needs. Alternatively, you can choose just the files that your application requires.

The library file `utils.lbr` contains the following support files:

If you require support for:	Files needed: (single-threaded)	Files needed: (multi-threaded)
CBL_EXEC_RUN_UNIT (graphical user interface applications)	<code>cbxecws.exe</code>	<code>cbxecwm.exe</code>
CBL_EXEC_RUN_UNIT (character user interface applications)	<code>cblexecs.exe</code>	<code>cblexecm.exe</code>
MF_CLIENT_STATE_* (Web application server-side persistence)	<code>sstate.gnt</code>	<code>sstate.gnt</code>
Character user interfaces (for true console applications only)	<code>cblvios.dll</code>	<code>cblviom.dll</code>
Character user interfaces (for applications that use the default COBOL text window)	<code>cbldwins.dll</code>	<code>cblwinm.dll</code>
File handling, for enhanced ACCEPT/DISPLAY syntax (ADIS), and for OO persistence	<code>mffh.dll</code>	<code>mffh.dll</code>
Running <code>.int</code> files, and for enhanced ACCEPT/DISPLAY syntax (ADIS)	<code>cblints.dll</code>	<code>cblintm.dll</code>
Making calls to the Win32 API from intermediate and generated code	<code>cob32api.dll</code>	<code>cob32api.dll</code>
PC_PRINT library routines	<code>cblprnt.dll</code>	<code>cblprnt.dll</code>

Other run-time support files that you might need to ship depending on the functionality that your application uses are as follows:

If you require support for:	You need to ship:
ActiveX controls:Toolbar	<code>comctl32.dll</code> (Microsoft module) <code>animlist.dll</code> <code>animpars.dll</code> <code>animserv.exe</code> <code>animsvr.lbr</code> <code>cbldbg.dll</code> <code>cobcli.lbr</code> <code>run.cfg</code>
Status bar	
Tree view	
Spin buttons	
Animator	

	run.exe
	runm.exe
	runmw.exe
	runs.exe
	runsw.exe
	runw.exe
	shell.lbr
	tools.lbr
	visremot.dll
Btrieve	xfh2btr.gnt
	_btrv.gnt
Command-line Sort	mfsort.exe
CCI:	
APPC protocol	cciap32.dll
DDE protocol	ccide32.dll
IPX protocol	ccix32.dll
NetBios protocol	ccinb32.dll
TCP/IP protocol	cciinst.exe and ccitc32.dll
TCP2 registration	ccitcp2.exe
Client/server binding	mfcsmgr.int
	mfserver.int
	mfclient.int
Common Gateway Interface (for deployment of a CGI server-side program on a Win32 platform)	
	acccgi.dll
	acccgi.int
Debugging remotely	cbldbg.dll
Dialog System (You need to include only those items your application uses)	
	cbldwins.dll
	cblints.dll
	cblrtss.dll
	cblvios.dll
	dsgrun.dll (Dialog System runtime)
	dsexcept.dll (Exception-reporting support)
	dsmdi.gnt (enables screensets to use MDI)
	dsrunner.gnt (DSRUNNER module)
	mfolex32.dll (OLE support)
	mfoxui32.dll (OLE support)
	oopss.dll
	pan2nt.dll
Dialog System extensions(You need to include only those extensions that your application uses.)(Note that if you use a dsdmi.gnt extension, you will need to link a dsdmi.obj to your application.)	
	dsbrowse.gnt

	dscnr.gnt
	dsdde.dll (DDE support)
	dsdir.gnt
	dsfw.lbr
	dsfwbw32.dll (Frame Window support)
	dsfwdf.lbr
	dsgrph32.dll (Graphics server)
	dsmovsz.gnt
	dsonline.gnt
	dsplayer.dll (DSPLAYER module)
	mfbps.dll (Graphics server)
	pvd40.ocx (ProtoView Grid Control)
	ActiveX controls (see top of table)
FaultFinder facility	cbldbg.dll
	cblffndm.exe (multi-threaded)
	cblffnds.exe (single-threaded)
File handling	xfh2btr.gnt (access to btrieve files)
Fileshare	fhrdrpwd.lbr
	fs.exe (server trigger)
	fsservice.exe (to run as an NT service)
	fsclose.exe (server closedown support)
	fsmgr.dll (manager API support)
Help - Character on-line help	helpname.lbr
	hyhelp.cfx
	hyhelp.hnf
	hyhelp.lbr
	hyhintf.gnt
	name.lbr
	utils.lbr
ISAPI server-side program (for deployment on a Win32 platform)	accisapi.dll
Java and COBOL mixed language support, depending on the JVM used by the application	
	cbljvm_ibm1.dll (IBM JVM V1.1)
	cbljvm_ibm2.dll (IBM JVM V1.2 or later)
	cbljvm_sun.dll (Sun JVM V1.2 or later)
	see also OO support
Micro Focus configuration (.cfg)	mfconfig.gnt

Micro Focus run trigger program:
multithreaded trigger
(Windows)single-threaded trigger
single-threaded trigger (Windows)

run.cfg
run.exe
runm.exe
runmw.exe
runs.exe
runsw.exe
runw.exe
shell.lbr
tools.lbr
utils.lbr

Microsoft Transaction Server

objectcontext.dll
drivers obtained separately, and
odbcw32.dll_sqlodbc.dll
pan2clib.gnt
pan2nt.dll
panels2.gnt

ODBC
Panels V2

Rebuild facility (command-line file recovery
tool)

callrb.lbr
rebuild.exe

mflangxx.lbr - these files need
to be available, and the
COBDIR environment variable
needs to point to the directory
containing them

SQL and ESQL

csqsupp.dll (Cobsq)
oesqlebc.dll (OpenESQL with
EBCDIC data)

Web Services support

apache-license.txt
castor-license.txt
castor-0_9_3_9-xml.jar
mfcc.exe
mfccmui.jar
muigen.gnt
xerces.jar

A3 - .int and .gnt Support Files to Ship

If your application contains .int files and .gnt files, you need to supply some files specifically for this type of application:

- The trigger program that starts your application. If do not supply your own trigger program, you can use one of the supplied trigger programs, run*.exe. If you do this, we strongly recommend installing the full Application Server run-time system using the supplied setup, and not shipping individual run-time files as described here.
- The run-time file that enables the run-time system to run .int files. This file is cblints.dll for single-threaded programs, and cblintm.dll for multi-threaded programs.
- The file for making calls to the Win32 API from intermediate and generated code: cob32api.dll.

In addition, you need the same range of files as any other application:

- The files that comprise your application, which in this case are .int and .gnt files.
- Any linked programs called by the application.
- The file for the shared run-time system: cblrtss.dll, for single-threaded applications, or cblrtsm.dll for multi-threaded applications

Any run-time support modules called by the application, or utils.lbr, which contains all the run-time support modules.

A4 - OO COBOL Support Files to Ship

If your application uses OO programming, you need to ship the following files:

If your application requires:	Files needed: (single-threaded)	Files needed: (multi-threaded)
OO programming support	class.dll classs.dll mfos.dll oopss.cfg oopss.dll	class.dll classm.dll mfom.dll oopsm.cfg oopsm.dll
Support for creating graphical user interfaces	apigui.dll apiguis.dll comctl32.dll (Microsoft module)	apigui.dll apiguim.dll comctl32.dll(Microsoft module)
Support for shipping programs as .int or .gnt files	cob32api.dll	cob32api.dll
OLE automation support	mfoles.dll oleclass.dll	mfolem.dll oleclasm.dll
OLE class library support	olecl.dll mfolecl.dll	olecl.dll mfolecl.dll
Java and COBOL mixed language support, depending on the JVM used by the application	cbljvm_ibm1.dll (IBM JVM V1.1) cbljvm_ibm2.dll (IBM JVM V1.2 or later) cbljvm_sun.dll (Sun JVM V1.2 or later) javaclass.dll (OO Java domain class library) mfcobol.jar (Java classes for accessing COBOL) mfjavam.dll (OO java domain)	

A5 - On-site Debugging Support Files to Ship

The following files are available for you to distribute to assist with on-site debugging, but we do not recommend that you ship them with your applications as a matter of routine:

cblanms.dll Support for animating shipped single-threaded executable files

cblanmm.dll Support for animating shipped multi-threaded executable files

cblcored.exe Support for core dump debugging of shipped executable files

B1 - How to set up a standalone application

Create a setup script to do the following:

- Install your application executable file.
- Search for an existing license database and set up a new one if necessary. See **C2 – How to set up a license database** section.
- Copy the remaining Application Server licensing files into the same directory as your application. See **A1 -Application Server licensing files** section
- Optionally, install the AS license key silently. See **C3 - How to install an AS license silently** section.

B2 - How to set up an application with its own Application Server

1. Create a setup script to install Application Server:
 - Copy the required Application Server files into the directory where you will install your application.
 - Search for an existing license database and set up a new one if necessary. See **C2 – How to set up a license database** section.
 - Optionally, install the AS license key silently. See **C3 – How to install an AS license silently** section.
2. Install the application executable files in the same directory as Application Server.
3. Create a shortcut pointing to the application. We recommend a shortcut in preference to adding the application directory to the user's path, so that you avoid interactions with other COBOL applications.
4. If the license database is installed on a server, set up the database location on the end-users' machines. See **C4 – How to set up a networked license database** section.

B3 - To set up an application with a shared Application Server

1. Install Application Server using the supplied Application Server setup program, which you can run interactively prompting the end-user for information, or run silently without the prompts. See **C1 – How to use the supplied Application Server setup** section.
2. Optionally, install the AS license key silently. See **C3 – How to install an AS license silently** section.
3. Install your application executable files in a separate directory from Application Server.
4. If Application Server is linked with dynamic binding, create a shortcut pointing to the application. We recommend a shortcut in preference to adding the Application Server directory to the user's path, so that you avoid interactions with other COBOL applications.

If Application Server is not linked with dynamic binding, set the PATH on the end-users' machines to point to Application Server, using the following command:

```
path = app-server-dir;%PATH%
```

where r:\app-server-dir is the directory where Application Server is installed. If you specify a directory containing spaces:

On Windows 98, you must enclose the directory in quotes. For example:

```
path = "r:\application directory";%PATH%
```

On all other Windows operating systems, you must not use quotes.

5. If the license database is installed on a server, set up the database location on the end-users' machines. See **C4 – How to set up a networked license database** section.

C1 - How to use the supplied Application Server setup

Run the Application Server setup program, setup.exe, which is in the application server directory of your Application Server for Net Express CD. This setup program is interactive and prompts the end-user for information. To run it, use the command:

```
setup
```

To install Application Server silently, without any prompts, use the -s option as follows:

```
setup -s
```

When you install silently, Application Server is installed in the default location and the license database is set up in its default location if there is no license database created yet. You can specify non-default locations, by specifying them in the silent install script file, setup.iss. This file is also in the application server directory of your Application Server for Net Express CD.

Specify the paths you want by editing the following lines of the script file:

In szPath, specify the directory where Application Server will be installed. For example, the following installs Application Server into the default directory:

```
szPath=C:\Program Files\Micro Focus\Application Server
```

In svPath, specify the directory where the license database will be set up if one does not already exist. For example, the following sets up the license database in its default directory:

```
svPath=C:\mfaslmf
```

If you do **NOT** want to be prompted for a serial and license key numbers, modify the file MicroFocusAppServ31.ini so that setup does not prompt for a license key. Set the entry to Yes:

```
[Requirements]
```

```
ASLMNoLicensePrompt=Yes
```

C2 - How to set up a license database

1. Find out whether a license database already exists, which might be the case if you have other applications using Application Server already installed. To do this, run the `aslmpclocate` utility as follows:

```
aslmpclocate [-q]
```

Where `-q` is optional and runs the utility quietly without interacting with the user.

Where the return code:

0	means ASLMFDIR is set
1	means ASLMFDIR is not set

If you are not running quietly, the location of the license database is displayed on the screen, or else "fail" is displayed together with the return code.

If you are running quietly, check the outcome using the return code, which is also stored in the registry (as a string containing a decimal number), in:

```
HKEY_LOCAL_MACHINE/SOFTWARE/Micro Focus/ASLMF/status
```

2. If there is no license database set up, add the following steps to your setup script:

Either prompt the user for a directory for the license database and create that directory, or use the default, which is `c:\mfaslmf`.

Copy the Application Server files `prodfile` and `semfile` into that directory.

Set up the database using the following command:

```
aslmpclocate [-q] path
```

Where:

<code>-q</code>	Is optional. Runs the utility in quiet mode without interacting with the user.
<code>path</code>	Is the path of the license database.

Where the return code:

0	means completed ok.
99	means other error

If you specify a path containing spaces, you do not need to enclose the path in quotes, although you can if you wish. For example, the following command sets up a database in the directory, `d:\my license database`:

```
aslmpclocate -q d:\my license database
```

3. If a license database already exists, you should normally use that one, but you can set up a new one and use that one instead, as described above. You will need to warn the user that the AS licenses in the existing database will no longer be accessible and that the licenses will need to be reinstalled in the new license database.

C3 - How to install an AS license silently

1. Set up the license database if you have not already done so. See **C2 – How to set up a license database** section.
2. Add a command to your application's setup script to install the license key silently, as follows:

```
aslmepsilent function serial license
```

Where the parameters must be in the above order and are:

function	Use 1 to install the license only if no other licenses are already installed. Use 2 to install the license, even if other licenses are already installed.
serial	Is the serial number, such as 112233449A. No spaces are allowed.
license	Is the license number, such as 01280 10000 018a5 53d0f 3fdd.

For example:

```
aslmepsilent 2 112233449A 01280 10000 018a5 53d0f 3fdd
```

3. If you will be using the supplied setup program to install Application Server, modify the file MicroFocusAppServ31.ini so that setup does not prompt for a license key. Set the entry to Yes:

```
[Requirements]
```

```
ASLMNoLicensePrompt=Yes
```

C4 – How to set up a networked license database

Set the ASLMFNET environment variable on the end-users' machines to point to the location of the network license database, as follows:

```
set ASLMFNET=license-db-path
```

Where license-db-path is the path of the license database. If you specify a path containing spaces, do not enclose the path in quotes.

Sample 1 “Silent” BAT installation file

```
echo off
setup -s
asImpclocate -q f:\myapsnx31\lcdb
asImpcsilent 1 NXDEVAS5 01280 10780 05C04 D5E09 55C0
echo Installation completed
```

Sample 2 “Silent” BAT installation file

```
echo off
echo ***** Starting installation of Application Server for Net Express *****
setup -s

asImpclocate -q

if errorlevel 0 goto FOUNDDDB
asImpclocate -q f:\myapsnx31\lcdb
asImpcsilent 1 NXDEVAS5 01280 10780 05C04 D5E09 55C0
if errorlevel 0 goto END
echo ***** License not set call Support *****
goto END

:FOUNDDDB
asImpcsilent 2 NXDEVAS5 01280 10780 05C04 D5E09 55C0
if errorlevel 0 goto END
echo ***** License not set call Support *****
goto END

:END
echo ***** Installation of Application Server for Net Express completed *****
```

Sample changes to be made to SETUP.ISS file used for silent install

```
[SdAskDestPath-0]
svPath=f:\myapsnx31\lcdb

[AskDestPath-0]
szPath=f:\myapsnx31
```

Sample MicroFocusAppServ31.ini file used for silent install

```
[Requirements]
ASLMNoLicensePrompt=Yes
```