

# Installing the Natural Web I/O Interface Server under SMARTS on z/VSE

This chapter describes how to install a server for the Natural Web I/O Interface (product code NWO) under the runtime environment SMARTS on z/VSE.

The installation of the Web I/O Interface server is performed by installation jobs. The sample jobs are contained in the dataset `NWOvrs.LIBJ`, or generated by System Maintenance Aid (SMA).

The following topics are covered:

- Prerequisites
- Content of the Web I/O Interface Server Distribution Tape
- Installation Procedure
- Allocating and Configuring a SMARTS Portable File System

## Prerequisites

For details, refer to the section *Prerequisites*.

## Content of the Web I/O Interface Server Distribution Tape

The installation tape contains the datasets listed in the table below. The sequence of the datasets and the number of library blocks needed are shown in the *Report of Tape Creation* that accompanies the installation tape.

Dataset Name	Contents
<code>APSVrs.LIBR</code>	Contains the load modules of the SMARTS server.
<code>NWOvrs.LIBR</code>	Contains the load modules of the Web I/O Interface server. See <i>Web I/O Interface Server Concept and Structure</i> .
<code>NWOvrs.LIBJ</code>	Contains Installation Job Control for customers who install without using System Maintenance Aid.

The notation `vrs` in the dataset names and in the program samples (below) represents the version, release and system maintenance level of the product.

## Installation Procedure

## Step 1: Create a Web I/O Interface server configuration file

(Job I009, Step 9400)

Define sublibrary for NWO environment.

(Job I009, Step 9410)

Catalogs the configuration file of the Web I/O Interface server. For a description of the parameters, refer to *Configuring the Natural Web I/O Interface Server*.

The following parameters of the configuration file must be defined. For the other parameters, the default values may be used:

FRONTEND_NAME	Specify the name of the server front-end module you will create in one of the following steps.
PORT_NUMBER	Specify the TCP/IP port number under which the server can be connected.

## Step 2: Create a SMARTS SYSPARM file

(Job I009, Step 9420)

Catalogs the configuration file SYSPARM for SMARTS.

For detailed information on the SMARTS configuration file, see *Configuration of the SMARTS Environment* in the SMARTS documentation.

(Job I009, Step 9430)

Create dummy member for NWO.

## Step 3: Link the object modules into the NWO load library

(Job I054, Step 9410)

The NWO object modules must be linked with the necessary runtime extensions of your batch installations into executable load modules.

## Step 4: Assemble and link reentrant ADALNK

Job I055, Step 9401, assemble and link reentrant ADALNK phase.

The server environment requires a reentrant ADALNK phase.

Link ADALNK using the ADALNKR module.

## Step 5: Assemble and catalog the Web I/O Interface server module

- Job I055, Step 9410, assemble and catalog the NCFNWOPM module.

## Step 6: Create the Web I/O Interface server front-end module

(Job I060, Steps 9410, 9420)

- Job I060, Step 9410, assemble and catalog the NATPARM module for NWO.
- Job I060, Step 9420, link the NWO front-end.

## Step 7: Catalog the SMARTS startup job

(Job I200, Step 9415)

Extend your SMARTS startup job by NWO-specific definitions described in the section *Configuring the Natural Web I/O Interface Server*.

### VSE Sample:

```
* $$ JOB JNM=NWOSRV,CLASS=C,DISP=L,LDEST=(,UID)
* $$ LST CLASS=A,DISP=H
// JOB NWOSRV --- NWO SERVER STARTUP ---
// OPTION PARTDUMP,NOSYSDMP,LOG
/* NWO server datasets -----
// DLBL  NWOSRVT,'/SAGLIB/NWOCNFG/NWOSRV.TRC'  writes trace to SAGLIB.NWOCNFG member NWOSRV.TRC
// DLBL  NWOSRVC,'/SAGLIB/NWOCNFG/NWOSRV.P'    location of NWO configuration file
// DLBL  NWOSRVE,'SYSLST'                      NWO error output directed to job output
// DLBL  NWOSRVO,'SYSLST'
// DLBL  SYSPARM,'/SAGLIB/NWOCNFG/MSGQ.DMY'    location of NWO Dummy file
// DLBL  STDOUT,'CONSOLE'                     STDOUT directed to VSE console
// DLBL  STDERR,'CONSOLE'
/* Libdef's -----
/*
// LIBDEF PHASE,SEARCH=(SAGLIB.NWOCNFG,          +
                      SAGLIB.APSvrs,          +
                      SAGLIB.ADAvrs)
/* *****
// UPSI  00000000
// EXEC  TLINSP,SIZE=AUTO
* $$ SLI MEM=RJANPARM.P,S=SAGLIB.APSvrs
* $$ SLI MEM=PXANCONF.P,S=SAGLIB.APSvrs
* $$ SLI MEM=SYSPARM.P,S=SAGLIB.NWOCNFG      NWO specific SMARTS configuration file
/*
// EXEC  LISTLOG
```

## Step 8: Web I/O Interface clients must be defined to Natural Security

If Natural Security (NSC) is installed:

- The Web I/O Interface initial user ID (default ID is STARGATE) must be defined in Natural Security with a valid default library. Refer also to Web I/O Interface server configuration parameter INITIAL\_USERID in the section *Configuring the Natural Web I/O Interface Server*. Alternatively, you can specify the Natural profile parameter AUTO=OFF (automatic logon) for the Web I/O Interface.
- Each client user ID must be defined in Natural Security.

If the Web I/O Interface initial user ID is not defined, the Web I/O Interface server initialization aborts with a NAT0856 error message.

If a Web I/O Interface client is not defined, the Map Environment dialog returns an NSC error.

If you connect to the server from a Web I/O Interface client, make sure that the user who is defined in Natural Security has a default library or a private library defined. Otherwise, the error message NAT0815 will be displayed.

## Step 9: Web I/O Interface clients must be defined to the server host

If you configure the Web I/O Interface server to use an external security system (see Web I/O Interface server configuration parameter `SECURITY_MODE`), the Web I/O Interface clients must be defined to the external security system.

# Allocating and Configuring a SMARTS Portable File System

The Natural server uses the SMARTS portable file system (PFS) as a data container for Natural work files, print files, temporary sort files and the editor work file. The SMARTS PFS is the only storage medium available for those files under SMARTS.

In order to be able to use the PFS for Natural files, you have to configure Natural accordingly:

- For work or print files, specify the access method `AM=SMARTS`, using the Natural profile parameters `WORK` and/or `PRINT`.
- For temporary sort files, specify the type of storage medium `STORAGE=SMARTS`, using the Natural profile parameter `SORT`.
- And to allocate the editor work file in the PFS, specify the work file mode `FMODE=SM`, using the Natural profile parameter `EDBP`.

If you use one of these options, you have to configure your SMARTS to use a PFS.

## Step 1: Allocate a PFS

Allocate a z/VSE file (`LRECL=4096`) with the estimated size (the file must start at cylinder boundary) to store your PFS, and completely initialize the container to contain `x'00's`.

The initialization can be done using the following job (runs until no more extents are available in the file and requires extra operator intervention to continue the job after the `WTOR` message for the extent overflow):

```
// JOB PFSFMT
// OPTION NODUMP
// DLBL CMWKF01,'your.pfs.file.name',0,SD
// EXTENT SYS001,volume,...
// ASSGN SYS001,DISK,VOL=volume,SHR
// ASSGN SYS000,READER
// UPSI 00000011
// LIBDEF PHASE,SEARCH=(SAGLIB.NATvrs)
// EXEC NATBATvr,SIZE=AUTO,PARM='SYSRDR'
IM=D,MADIO=0,MT=0,OBJIN=R,ID=',',INTENS=1,
WORK=( (1),AM=STD,RECFM=FB,BLKSIZE=4096,LRECL=4096)
EDT
DEFINE DATA LOCAL
1 B(B1) INIT<H'00'>
END-DEFINE
```

```

DEFINE WORK FILE 1 'CMWKF01'
REPEAT
WRITE WORK 1 B
WHILE 1=1
END
.E
RUN
FIN
/*

```

## Step 2: Configure SMARTS

In the SMARTS startup JCL, add the following statements:

```

// DLBL APSPFS1,'your.pfs.file.name',0,DA
// EXTENT SYSvrs,volume,...
// ASSGN SYSvrs,DISK,VOL=volume,SHR

```

In the SMARTS SYSPARM file, add the following statements (where *vrs* stands for the current Natural version, release number and system maintenance level):

```

RESIDENTPAGE=NCFvrsAPS
CDI_DRIVER=( 'CIO,PAANCIO' )
CDI=( 'PFS1,PAANPFS,CACHESIZE=2000,LRECL=4096,CONTAINER=CIO://DD:APSPFS1' )
MOUNT_FS=( 'PFS1://', '/usr/' )
ENVIRONMENT_VARIABLES=/SAGLIB/APSvrs/ENVARS.P

```

Create a member ENVARS.P under SAGLIB.APSvrs containing the following lines:

```

NAT_WORK_ROOT=/usr/natural/etc/work_file
NAT_PRINT_ROOT=/usr/natural/etc/print_file
NCFWFAPS_TRACE_LEVEL=0

```

NCFWFAPS\_TRACE\_LEVEL=31 may be used for diagnostic purposes. It causes all PFS accesses to be traced by Natural and to be written to the Web I/O Interface server dataset STDOUT. It is recommended to set the NCFWFAPS\_TRACE\_LEVEL parameter value to zero.

## Step 3: Verify PFS Configuration

SMARTS now should protocol the following line in SYSLST during startup:

```
APSPSX0050-SYSNAME CDI PFS1 PROTOCOL INITIALIZED
```

Once your Natural Web I/O Interface server installation has been completed, issue a **Map Environment** command to the Natural Web I/O Interface server, using the following session parameter:

```
WORK=( ( 1 ), AM=SMARTS )
```

The following Natural program should run and display Record 01:

```
0010 DEFINE DATA LOCAL
0020 1 A(A20)
0030 END-DEFINE
0040 A := 'Record 01'
0050 WRITE WORK 1 A
0060 CLOSE WORK 1
0070 READ WORK 1 A
0080 WRITE A
0090 END-WORK
0100 END
```