

# Installing and Configuring the Natural Web I/O Interface Server

On UNIX, the server part of the Natural Web I/O Interface is implemented as a daemon.

This chapter covers the following topics:

- Installing the Natural Web I/O Interface Daemon
  - Before You Start
  - Setting Up the Natural Web I/O Interface Components
  - Directories
  - Configuring the Natural Web I/O Interface Daemon on UNIX
  - Logging Information
  - SSL Support
  - Working with the UNIX Components of the Natural Web I/O Interface
- 

## Installing the Natural Web I/O Interface Daemon

The installation of the Natural Web I/O Interface daemon is part of the Natural for UNIX installation.

### Before You Start

This section contains important information on the necessary activities before installing Natural Web I/O Interface daemon.

The Natural Web I/O Interface daemon `$SAG/nat/$NATVERS/nwo/bin/nwosrvd`

- needs a Tcl shared library which is delivered in the directory `$SAG/nat/$NATVERS/lib`,
- is linked with the runpath `/opt/softwareag/nat/$NATVERS/lib`,
- will be installed with permissions `6755` (s-bit).

Since the s-bit is used, `$LD_LIBRARY_PATH` will not be searched. Therefore, ensure that the Natural Web I/O Interface daemon will find the Tcl shared library by

- installing Natural into `/opt/softwareag`,
- setting a symbolic link from `opt/softwareag` to your current `$SAG` directory, or
- making the Tcl shared library available from a system directory.

# Setting Up the Natural Web I/O Interface Components

Setting up the Natural Web I/O Interface on UNIX consists of the following steps:

- Step 1: Stop the Natural Web I/O Interface Daemons
- Step 2: Establish the Environment
- Step 3: Install Natural and the Natural Web I/O Interface
- Step 4: Check the Environment Variables for the Natural Web I/O Interface
- Step 5: Read the READ\_NWO Files

## Step 1: Stop the Natural Web I/O Interface Daemons

This step is only required for an upgrade installation. It is not required when you install the Natural Web I/O Interface for the first time.

1. Stop the *nwosrvd* process using the following command:

```
nwosrvd.sh portnumber stop
```

Or use the script *\$NATDIR/\$NATVERS/INSTALL/nwosrvd.bsh* which will be generated during the Natural Web I/O Interface installation for a specified port.

```
nwosrvd.bsh stop
```

2. Repeat the above command (with an adapted port in script *nwosrvd.bsh*, if applicable) for each Natural Web I/O Interface service that is needed.

## Step 2: Establish the Environment

- Ensure that the environment settings in the file *sagenv.new* are correct and set. Note that the *nwoenv* environment script will be called by the *natenv* environment script.

Or use the shell script *nwoenv.csh* by entering the following command:

```
source nwoenv.csh
```

This script can be found after the installation in *\$NATDIR/\$NATVERS/INSTALL*.

### Step 3: Install Natural and the Natural Web I/O Interface

- The Natural Web I/O Interface can be selected in the **Choose Packages** screen during the Natural installation.

Optionally, you may install a runlevel script to start/stop a Natural Web I/O Interface daemon and start the Natural Web I/O Interface daemon on a specified port. After the Natural installation has finished, the Natural Web I/O Interface must be activated by starting Natural through a Natural Web I/O Interface client on Windows.

When a runlevel script is used, the Natural Web I/O Interface daemon can only be administered by the user "root".

When you install Natural with the Natural Web I/O Interface, the directory *\$NATDIR/nwo/\$NWONODE* is created. The template files located in *\$NATDIR/\$NATVERS/nwo/node-name* are then copied to this new directory.

### Step 4: Check the Environment Variables for the Natural Web I/O Interface

- The Natural Web I/O Interface-specific settings are shown below:

Environment Variable	Description
NWODIR	The home directory for the product located at <i>\$NATDIR/\$NATVERS/nwo</i> .
NWONODE	The name of the node on which the Natural Web I/O Interface is installed.
NWO_SRVDCONF	The configuration file <i>\$NATDIR/nwo/\$NWONODE/nwosrvd.conf</i> for the Natural Web I/O Interface daemon.
NWO_TIMEOUT	The maximum time, in seconds, that the Natural Web I/O Interface daemon will wait for a response. "0" means no timeout. The Natural Web I/O Interface daemon will terminate when it receives the timeout.

### Step 5: Read the READ\_NWO Files

1. Access the directory *\$NATDIR/\$NATVERS* and check the files *READ\_NWO.TXT* and *READ\_NWO.FIX* for any version-specific installation considerations concerning the particular platform.
2. Add the services as described in the file *READ\_NWO.TXT*.

## Directories

The following directories are created when Natural is installed together with the Natural Web I/O Interface on a UNIX system:

Directory	Description
<i>\$NATDIR</i>	Top-level Natural directory.
<i>\$NATDIR/\$NATVERS</i>	Directory with all components for the current Natural version.
<i>\$NWODIR</i>	Directory with the Natural Web I/O Interface components for the current version.
<i>\$NWONODE</i>	Contains the name of the machine ( <code>uname -n</code> ).
<i>\$NATDIR/\$NATVERS/INSTALL</i>	Shell scripts and environment files for the Natural Web I/O Interface ( <i>nwoenv</i> , <i>nwoenv.csh</i> ).
<i>\$NWODIR/bin</i>	Natural Web I/O Interface executable files ( <i>nwosrvd</i> , <i>nwosrvd.tr</i> ).
<i>\$NWODIR/node-name</i>	Contains the template files ( <i>nwosrvd.sh</i> , <i>nwo.sh</i> , <i>nwosrvd.conf</i> ).
<i>\$NWODIR/nwoexuex/userexit1</i>	Contains the files for building the <i>libnwouserexit1</i> .
<i>\$NWODIR/nwoexuex/userexit2</i>	Contains the files for building the <i>libnwouserexit2</i> .
<i>\$NATDIR/nwo/\$NWONODE</i>	Work directory, contains the configuration files ( <i>nwosrvd.sh</i> , <i>nwo.sh</i> , <i>nwosrvd.conf</i> ).

**Note:**

The above table lists the most important directories and files.

## Configuring the Natural Web I/O Interface Daemon on UNIX

When the Natural installation has finished, the directory *\$NATDIR/nwo/\$NWONODE* contains the files *nwosrvd.conf*, *nwosrvd.sh* and *nwo.sh*.

The configuration of the Natural Web I/O Interface daemon can be done using the Natural Web I/O Interface daemon commands or by editing the configuration file *nwosrvd.conf*.

The following topics are covered below:

- Natural Web I/O Interface Daemon Commands
- *nwosrvd.conf* - Configuration File for the Natural Web I/O Interface Daemon
- *nwosrvd.sh* - Shell Script for Starting and Stopping the Natural Web I/O Interface Daemon
- *nwo.sh* - Shell Script for Starting Natural
- Environment Variables

### Natural Web I/O Interface Daemon Commands

The following commands can be specified at the UNIX command prompt:

<b>Command</b>	<b>Description</b>
<code>nwosrzd -help</code>	Shows all available Natural Web I/O Interface daemon commands and subcommands.
<code>nwosrzd -v</code>	Shows the version of the Natural Web I/O Interface daemon.
<code>nwosrzd nnnn</code>	Defines the listening port number.
<code>nwosrzd -show</code>	Shows the configuration of the Natural Web I/O Interface daemon.

Command	Description
<pre>nwosrvd -config keys</pre>	<p>Changes the configuration of the Natural Web I/O Interface daemon. The following keys can be specified:</p> <p><b>-userexit1=pathname</b></p> <p>The message defined with this key is saved in the <code>UserExit1</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[UserExits]</code>.</p> <p><b>-userexit2=pathname</b></p> <p>The message defined with this key is saved in the <code>UserExit2</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[UserExits]</code>.</p> <p><b>-passparam=parameters</b></p> <p>The message defined with this key is saved in the <code>Parameters</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[PasswdArguments]</code>.</p> <p><b>-passold=message</b></p> <p>The message defined with this key is saved in the <code>EnterOldPassword</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[PasswdMessages]</code>.</p> <p><b>-passnew=message</b></p> <p>The message defined with this key is saved in the <code>NewPassword</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[PasswdMessages]</code>.</p> <p><b>-passreenter=message</b></p> <p>The message defined with this key is saved in the <code>ReEnterNewPassword</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[PasswdMessages]</code>.</p> <p><b>-passsuccess=message</b></p> <p>The message defined with this key is saved in the <code>PasswordSuccessful</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[PasswdMessages]</code>.</p> <p><b>-logging=option</b></p> <p>The option defined with this key is saved in the <code>Logging</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[Miscellaneous]</code>.</p> <p><b>-ssl=[yes no]</b></p> <p>The option defined with this key is saved in the <code>ssl</code> key of the configuration file <code>nwosrvd.conf</code>, section <code>[SSL]</code>.</p> <p>To remove any user exits from the configuration, enter the following command:</p> <pre>nwosrvd -config -userexit1=</pre> <p>Once the configuration was changed, the Natural Web I/O Interface daemon must be restarted.</p>

## ***nwosrvd.conf* - Configuration File for the Natural Web I/O Interface Daemon**

The configuration file *nwosrvd.conf* contains information that the user exits need for the Natural Web I/O Interface daemon. It has the following content:

```
[Miscellaneous]
Logging=I

[UserExits]
; UserExit1=/FS/sag/nat/nwoexuex/userexit1/libnwouserexit1.so
; UserExit2=/FS/sag/nat/nwoexuex/userexit2/libnwouserexit2.so

[PasswdArguments]
Parameters=

[PasswdMessages]
EnterOldPassword=Enter existing login password:
NewPassword=New Password:
ReEnterNewPassword=Re-enter new Password:
PasswordSuccessful=passwd: password successfully changed for*

[SSL]
ssl=no
```

Section in Configuration File	Description
[Miscellaneous]	<p>The key <code>Logging</code> is used to define the amount of logging information that is to be reported. One of the following options can be specified:</p> <p>E for errors.  W for warnings.  I for information.</p> <p>See also <i>Logging Information</i>.</p>

Section in Configuration File	Description
[UserExits]	<p>Two user exits can be defined:</p> <p><b>UserExit1</b></p> <p>The library that is defined by UserExit1 contains the following function:</p> <pre>int nwo_CheckUsernameAndPassword(const char *pUsername, const char *pPassword, const char *pNewPassword, char *pErrorMessage)</pre> <p>If the key UserExit1 is defined in the configuration file, the function nwo_CheckUsernameAndPassword is responsible for checking the user name and password. If a new password is received, user exit 1 is also responsible for changing the password.</p> <p>In the case of an error, the return code of the function must be "0"; in this case, the pErrorMessage is returned to the client. When user name and password are correct, the return code must be a value other than "0".</p> <p><b>UserExit2</b></p> <p>The library that is defined by UserExit2 contains the following functions:</p> <ul style="list-style-type: none"> <li>● <code>int nwo_Messages(int *iNumberOfMessages, char *pMessage[])</code></li> </ul> <p>iNumberOfMessages: Number of messages returned in the array.</p> <p>pMessage: Array of messages.</p> <p>If the key UserExit2 is defined in the configuration file, the function nwo_Messages is called when a new connection (client) is accepted and the messages returned by this function are sent to the client. User exit 2 may be used, for example, to send a message such as the following: "For maintenance reasons, the Natural application XXXXX will be down next monday, from 18:00 until 19:00".</p> <p>In the case of an error, the return code of the function must be "0".</p> <p>After the function nwo_Messages has been called, the function nwo_FreeMessages is called.</p> <ul style="list-style-type: none"> <li>● <code>int nwo_FreeMessages(int iNumberOfMessages, char *pMessage[])</code></li> </ul> <p>iNumberOfMessages: Number of messages.</p> <p>pMessage: Array of messages.</p> <p>If the key UserExit2 is defined, the function nwo_FreeMessages is called to free any resources (normally memory) allocated in the function nwo_Messages.</p> <p>In the case of an error, the return code of the function must be "0".</p>
[PasswdArguments]	The key Parameters is used to define any additional parameter(s) that have to be passed to the passwd command.



Section in Configuration File	Description
[ PasswdMessages ]	<p>The keys in this section define the messages that are to be returned by the system (<code>passwd</code> command) when a user changes the password. If any of these messages is not identified by the daemon, an error will be returned to the client.</p> <p><b>Password Mechanism</b></p> <p>The password and new password are encrypted on the client side and decrypted on the UNIX side. A maximum of 8 characters is allowed.</p> <p>If user exit 1 is active, user name, password and new password are passed to the user exit.</p> <p>If user exit 1 is not active, the daemon checks whether user name and password are correct for the system. If a new password is sent, the daemon changes the password by calling the UNIX command <code>passwd</code>.</p>
[ SSL ]	<p>The key <code>ssl</code> is used to define whether the SSL protocol is to be used. One of the following values can be specified: "yes" or "no".</p> <p>See also <i>SSL Support</i>.</p>

## ***nwosrvd.sh* - Shell Script for Starting and Stopping the Natural Web I/O Interface Daemon**

The shell script *nwosrvd.sh* is used to start and stop the Natural Web I/O Interface daemon. For further information, see *Starting and Stopping the Natural Web I/O Interface Daemon*.

## ***nwo.sh* - Shell Script for Starting Natural**

In order to start a Natural session, the Natural Web I/O Interface service executes a shell script. The shell script prepares the environment for the Natural session and eventually starts Natural.

The shell script receives certain parameters from the Natural Web I/O Interface client. The parameters can either be evaluated by the shell script itself or passed on to Natural. A client who wants to start a Natural session can specify the shell script to be used.

The shell script *nwo.sh* is called from the Natural Web I/O Interface daemon in order to start a Natural session. It has the following content:

```
#!/bin/sh

echo "Number of arguments $# " > nwo.log

IPAddress=""
ClientId=""
CodePage=""
CustomParameters=""
NaturalParameters=""

if [ "$1" != "null" ]
then
    IPAddress="$1"
fi

if [ "$2" != "null" ]
```

```
then
  ClientId="$2"
fi

if [ "$3" != "null" ]
then
  CodePage="$3"
fi

if [ "$4" != "null" ]
then
  CustomParameters="$4"
fi

if [ "$5" != "null" ]
then
  NaturalParameters="$5"
fi

echo "IP Address="$IPAddress >> nwo.log
echo "Client Id="$ClientId >> nwo.log
echo "Code Page="$CodePage >> nwo.log
echo "Custom Parameters="$CustomParameters >> nwo.log
echo "Natural Parameters="$NaturalParameters >> nwo.log
echo "NWO_BROWSER_IO="$NWO_BROWSER_IO >> nwo.log

$NATDIR/$NATVERS/bin/natural $NaturalParameters etid=$$ > /tmp/natural_$$$.out 2>&1
```

You have to create such a shell script for each Natural application. It can have any name and it must be located in an directory which is defined in the environment variable PATH.

The name of the shell script is taken from the configuration file for the session. It is taken from the configuration file section that is defined for the session that the user has selected in the logon page. For further information, see *Configuring the Client*.

## Arguments

The shell script will receive the following arguments:

Order	Argument	Description
1	IPAddress	The client IP address from where the session is opened.  <b>Note:</b> If there is a proxy, this will not be the IP address of the client workstation. Instead, it will be the IP address of the proxy.
2	ClientId	The user name from the logon page is passed as the client ID.
3	CodePage	The encoding that is defined in the configuration file for the session. This value can be used to set the Natural system variable *CODEPAGE.
4	CustomParameters	From the logon page, it is possible to pass any values to the script in order to execute any desired action.  Example: you pass a small text to the script which describes an error. When the script receives this error text, it sends it as an e-mail to the administrator.
5	NaturalParameters	These can be any Natural parameters. The parameters are either defined in the configuration file for the session, or they are entered in the logon page. The following is an example of the corresponding entry in the configuration file:  <pre>&lt;natural_parameter&gt;parm=nwoparm\ stack=(logon\ mylib;start-program;fin)&lt;natural_parameter&gt;</pre> The language that is selected in the logon page is added as the first element to the Natural parameters in the form "ulang=x".

Arguments 1 to 4 can be used to audit the client, to allow to run an application from a specific PC (identifying the IP address), to build statistics, to do special actions, etc.

## Environment Variables

In the shell script, several environment variables can be set for the Natural session that is started by the daemon:

**NWO\_ENABLE\_ACK=["YES"|"NO"]**

This environment variable is used for asynchronous screens (SET CONTROL N).

**YES** When asynchronous screens are sent to the client, Natural will wait to receive an ACK package before the next screen can be sent.

**NO** No waiting between asynchronous screens. Default value.

**NWO\_TIMEOUT=[*number-of-seconds*]**

The maximum time, in seconds, that Natural waits to receive any input from the client before it closes the session. If the number of seconds is "0", Natural waits infinitely (no timeout). The default value is "0".

Error NAT5466 is returned at timeout. In Natural, the application can handle this error and decide how to continue or terminate.

## Logging Information

The logging information system reports errors, warnings and/or session information, depending on the option that has been defined with the following Natural Web I/O Interface daemon command:

```
nwosrvd -config -logging=option
```

*option* can be one of the following:

Option	Description
E	<p>Error.</p> <p>When this option is specified, the Natural Web I/O Interface daemon reports only errors.</p> <p>In the case of an error, the daemon usually exits immediately.</p>
W	<p>Warning.</p> <p>When this option is specified, the Natural Web I/O Interface daemon reports errors and warnings for uncritical errors.</p> <p>In the case of a warning, the daemon continues to run.</p>
I	<p>Information.</p> <p>When this option is specified, the Natural Web I/O Interface daemon reports errors, warnings and information.</p> <p>The information messages allow to check the session parameters, IP address, etc.</p>

Help information, for example, on how to run, configure and install the Natural Web I/O Interface daemon is always provided. The messages which inform you when the daemon has been started or stopped are also part of the help information.

To find out which logging option is currently active, enter the following Natural Web I/O Interface daemon command:

```
nwosrvd -show
```

The logging messages are shown directly for the standard output. The format of the messages is as in the following example:

```
%NWOSRVD-E: 18.01.2008 14:55:20 NWO_SRVDCONF is not established.
```

The following information is provided:

- %NWOSRVD is the internal name of the Natural Web I/O Interface daemon.
- The message type is shown directly after %NWOSRVD. It can be one of the following: -E (error), -W (warning), -I (information), or -H (help).
- Date and time when the message was reported.
- Any text or message which pertains to the error, warning, information or help.

If you want to save these messages, you have to redirect the standard output to a file.

Example for csh:

```
nwosrvd 5454 >& nwosrvd_5454.log
```

Example for sh, ksh and bsh:

```
nwosrvd 5454 >& nwosrvd_5454.log 2>&1
```

## SSL Support

SSL is used for a secure connection between the Natural Web I/O Interface server and the Natural Web I/O Interface client or Natural for Ajax. Server authentication cannot be switched off. A certificate and a private key is always required on the server.

To create and use an SSL certificate and a private key on the server, proceed as described below.

- Adapt the example configuration file *openssl.cnf* to your needs.

**Note:**

*openssl.cnf* and *openssl* are delivered in *\$NATDIR/\$NATVERS/bin*.

- Set the environment variable so that it points to the file *openssl.cnf*:

```
set OPENSSL_CONF=$NATDIR\NATVERS\bin\openssl.cnf
export OPENSSL_CONF;
```

- Generate a certificate signing request:

```
openssl req -new > server.cert.csr
```

- Generate a private RSA key:

```
openssl rsa -in privkey.pem -out server.cert.key
```

- Generate a self-signed certificate:

```
openssl x509 -in server.cert.csr -out server.cert.crt -req -signkey server.cert.key -days 365
```

**Note:**

The certificate can be self-signed or it can be signed by a CA (Certificate Authority) such as VeriSign.

- Put the generated files into the same directory as the scripts which start the Natural Web I/O Interface server.
- Import the generated files to a truststore on the client. See also *Trust Files (J2EE only)*.

## Working with the UNIX Components of the Natural Web I/O Interface

The UNIX components of the Natural Web I/O Interface are used to start the Natural applications linked with the Natural Web I/O Interface library.

The following topics are covered below:

- Starting and Stopping the Natural Web I/O Interface Daemon
- Starting a Natural Application

### Starting and Stopping the Natural Web I/O Interface Daemon

The Natural Web I/O Interface daemons are responsible for accepting new sessions.

Since the daemon checks the user name and password, the following permissions must be set as follows (for setting the permissions, you must be super-user):

```
chmod 6755 nwsrsvd.sh
```

```
chown root nwsrsvd.sh
```

The Natural installation attempts to set permissions and owner. However, you have to verify this before you start the Natural Web I/O Interface daemon.

The daemon can be started and stopped using the following command:

```
cd $NATDIR/nwo/$NWONODE
nwosrvd.sh portnumber [start|stop]
```

Alternatively:

```
cd $NATDIR/$NATVERS/INSTALL
nwosrvd.bsh [start|stop]
```

**Note:**

The daemon must be started on a port which is not yet used.

The shell script you have created must be in the same directory as the *nwosrvd.sh* script. It will be used by the Natural Web I/O Interface (configuration file for the session; see *Configuring the Client*). The following is an example of the corresponding entry in the configuration file:

```
<natural_program>your-shell-script.sh</natural_program>
```

## Starting a Natural Application

Almost any Natural application can be used with the Natural Web I/O Interface. See also *Differences between the Natural Web I/O Interface Client and Terminal Emulation*.

To start a new Natural application with the Natural Web I/O Interface, proceed as follows:

1. Create a new parameter file from NWOPARM using the Configuration Utility.
2. In this new parameter file, modify the STACK command as follows:

```
logon library; startprogram; fin
```

**Note:**

Only "real" Natural applications can be used. The Natural **Main Menu** cannot be used as a Natural application.

Add the new service as follows:

1. Look for a port number which is not yet used.
2. Create a new shell script (similar to *nwo.sh*) for starting the Natural application:

```
cd $NATDIR/nwo/$NWONODE
copy nwo.sh your-shell-script.sh
vi your-shell-script.sh
```

You have to decide which (last) line you will use in the script. Use one of the following:

```
$NATDIR/$NATVERS/bin/natural parm=parameter-file etid=$$ >output-file 2>&1
```

```
$NATDIR/$NATVERS/bin/natural $5 etid=$$ >output-file 2>&1
```

When using the line with `parm=parameter-file`, the above step in which you modify the `STACK` command is mandatory.

When using `$5`, the Natural parameter (*parameter-file* and `STACK` command) is taken from the configuration file for the session (see *Configuring the Client*). The following is an example of the corresponding entry in the configuration file:

```
<natural_parameter>parm=myparm stack=(logon mylib;menu;fin)<natural_parameter>
```

3. If you want to define special settings for the Natural session, you can set the environment variables in your shell script. See above.
4. Set the permissions for the shell script which starts the service as follows:

```
chmod 775 script-name
```

The service is now available for use with a PC.