



Introduction

Welcome to the FACTS 7 Installation Manual. This guide is provided to help you through the FACTS installation and configuration process.

Installing the FACTS software is just one step in the full implementation of the FACTS system. The Implementation Methodology, *Partnership in Action Implementation Workbook*, is a structured process that Affiliates and Implementation Consultants should follow. Customers should be given the *Partnership in Action Customer Implementation Planning Guide* to help them complete their implementation responsibilities.

Installation prerequisites

You must be familiar with UNIX and NT operating systems to successfully install the FACTS system and you should know how to configure TCP/IP in the customer's chosen operating system.

You will also need a working knowledge of the BASIC language to run ProvideX conversion programs and utilities.

Understanding the FACTS directory structure

Once installed on the server, FACTS resides in a straightforward, well-organized directory structure comprised of data files and program files.

The FACTS installation media is set up so that the installation process creates this structure in a FACTS subdirectory named **ssi7** in UNIX or **SSI7** in DOS, which becomes the main FACTS directory. Determine where on your server that main directory needs to reside and indicate that during the installation procedure.

For example, if you install FACTS in the root directory, the main FACTS directory becomes **/ssi7** in UNIX or **C:\SSI7** in NT.

The following tables illustrate the main FACTS directory (**ssi7**) and its subdirectories as they appear on the UNIX installation media. The NT (DOS) directory structure is the same, except it uses backslashes.

Core Modules	FACTS Data Files	FACTS Programs	Other
Accounts Payable	ssi7/data/AP	ssi7/prog/AP	
Accounts Receivable	ssi7/data/AR	ssi7/prog/AR	
General Ledger	ssi7/data/GL	ssi7/prog/GL	
Inventory Control	ssi7/data/IC	ssi7/prog/IC	
Purchase Orders	ssi7/data/PO	ssi7/prog/PO	
Sales Analysis	ssi7/data/SA	ssi7/prog/SA	
System Management	ssi7/data/SM	ssi7/prog/SM	
Office Automation	ssi7/data/OA	ssi7/prog/OA	
Job Stream	ssi7/data/JS	ssi7/prog/JS	
Sales Orders	ssi7/data/SO	ssi7/prog/SO	
Add-on Modules	FACTS Data Files	FACTS Programs	Other
CodeLight	ssi7/data/CL	ssi7/prog/CL	*additional files
FaxLink*	ssi7/data/FL	ssi7/prog/FL	ssi7/faxlink
Job Cost	ssi7/data/JC	ssi7/prog/JC	
Manufacturing Control	ssi7/data/MC	ssi7/prog/MC	
Payroll	ssi7/data/PR	ssi7/prog/PR	
Service and Repair	ssi7/data/SR	ssi7/prog/SR	
TeleFACTS	ssi7/data/TF	ssi7/prog/TF	

Chart continued on next page

Other	FACTS Data Files	FACTS Programs	Other
ProvideX			/ssi7/pvx/
WindX			\pvx
ODBC			
Adobe Acrobat			\ssi7\Acrobat
FacetTerm*			usr/faceterm
Report Writer		ssi7/prog/RW	ssi7/gen4x
Unform			ssi7/unform
VSI-FAX			ssi/vsi-fax
Modification Support			ssi7/mods

* FaxLink, FacetTerm and FacetWin are only available on UNIX systems.

Understanding the FACTS Authorization Code Sheet

This is a confidential document Software Solutions, Inc. delivers with each new installation. It provides you with information that is critical to the installation process.

- FACTS serial number
- Number of users licensed for FACTS
- Primary company name
- Modules purchased
- List of companies that will be set up in FACTS
- Serial numbers and activation keys for all ProvideX products purchased

➡ **If you did not receive an Authorization Code Sheet or if the information on the sheet is incorrect, call Software Solutions, Inc. at (800) 774-0233.**

Because the order in which you enter this information is critical, please make sure you are familiar with the sheet's layout and its contents before you begin.

Look at the sample document show on the next page and notice that the document essentially consists of two parts. The top section contains FACTS authorization information. Beneath that, you'll find serial numbers and activation keys for ProvideX products.

Sample Confidential FACTS Authorization Code Sheet

Page 1- 02/09/99

FACTS Serial #: 000111

Max FACTS Users: 8

Main Company: OMNE DISTRIBUTION

Authorized

Modules: AP AR SO PO IC GL SA SM

Comp Code	Company Name	Authorization Code
01	OMNE DISTRIBUTION	aa1BcD2EfGh
99	DEMO COMPANY	

The top portion contains FACTS authorization information that needs to be entered when setting up System Control F/M and Company F/M in the INSTALL menu.

Product	PVX Serial #	Users	Activation Key	Exp Date
ProvideX	001122	10	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	
WindX	000000	1	A1234B5678D	

Though listed after the FACTS information, the ProvideX serial number and activation key are each entered at the beginning of the installation process.

Make sure you **do not** enter a WindX serial number or activation key by accident.

*** Legal Notice *** THESE CODES ARE TO BE USED ONLY IN COMPLIANCE WITH THE TERMS OF THE SOFTWARE LICENSING AGREEMENT

*** Important Note *** Your serial number - 000111 - will also be used by your local Software Solutions representative to receive support, updates and enhancements on your behalf while you are covered by the Software Solutions Maintenance Policy.

Understanding FACTS Security

The FACTS security system is built into the menu driver so that it can monitor the number of users logging into FACTS versus the number of licenses purchased.

It relies on an environment variable, called SSI_BASE, and an SMFIDS file to cross reference FID(0), which is the value that appears as Terminal ID in System Management's Terminal F/M.

SSI_BASE is the heart of the security system, enabling it to determine who is logging into FACTS and how many times. The variable also compares this information to the number of licenses purchased.

➤ **Each licensed user can log into FACTS up to nine times, as long as they do so with the same FACTS user code and on the same workstation.**

In any SSI_BASE discussion the word *unique* appears often. How FACTS uses SSI_BASE is the reason why administrators shouldn't rely on pseudo ttys as SSI_BASE values and why SSI_BASE needs to be recognized as early as possible in the start up procedure.

When users log into FACTS, the security system captures the SSI_BASE value and SME100 checks the SMFIDS file for the SSI_BASE to determine and set the Terminal ID. These Terminal IDs remain in SMFIDS as long as the User ID is maintained. Every time a user starts a session, the first session will be Terminal ID TA, for example; the second session will be terminal ID TB and so on.

The following table shows a NT/PC user, JohnD, who successfully signed into FACTS three times. He can log into the system six more times with the user code "SSI" without experiencing any problems.

	<u>SSI_BASE</u>	<u>If UID OK, check NID</u>	<u>If NID OK, check TCP/IP</u>	<u>If TCP/IP OK, check C0\$(1,3)</u>	<u>If C0\$(1,3) OK, check ...</u>
Session 1	john	UID=JohnD	NID=Iron	TCP/IP=123.1.1.1	User code=SSI
Session 2	john	UID=JohnD	NID=Iron	TCP/IP=123.1.1.1	User code=SSI
Session 3	john	UID=JohnD	NID=Iron	TCP/IP=123.1.1.1	User code=SSI

When the user launched the first session, the FACTS security system noted network signon (UID), the PC's network ID (NID), the PC's TCP/IP address and the FACTS user code.

As new sessions are launched, the security system checks each of these identifiers in succession. As soon as one fails to match, it drops all sessions for that SSI_BASE.

UNIX installations require SSI_BASE to be set manually. Instructions are provided in Chapter 2.

The NT server and client installation procedures automatically sets SSI_BASE.

In the following table, for instance, one user launched two FACTS sessions successfully. However, the third attempt failed because the security system detected a network signon for SSI_BASE=john that differs from the first two sessions. This indicates that same SSI_BASE value may have been set in two different autoexec.bat files.

	<u>SSI_BASE</u>	If UID OK, check NID	If NID OK, <u>check TCP/IP</u>	If TCP/IP OK, <u>check C0\$(1,3)</u>	If C0\$(1,3) <u>OK, ...</u>
Session 1	john	JohnD	Iron	123.1.1.1	SSI
Session 2	john	JohnD	Iron	123.1.1.1	SSI
Session 3	john	MaryF	Quartz	127.3.3.3	SSI

The next example shows a user trying to launch a third FACTS session with a different FACTS user code. His third attempt will be successful, but when he returns to the other two sessions and tries to open a program, "User Not Signed On, CR-Continue" will display at the bottom of the screen.

	<u>SSI_BASE</u>	If UID OK, check NID	If NID OK, <u>check TCP/IP</u>	If TCP/IP OK, <u>check C0\$(1,3)</u>	If C0\$(1,3) <u>OK, ...</u>
Session 1	john	JohnD	Iron	123.1.1.1	SSI
Session 2	john	JohnD	Iron	123.1.1.1	SSI
Session 3	john	JohnD	Iron	123.1.1.1	MAR

The System Management module also includes a System Security menu that enables administrators to establish user and program level security systems in addition to the login security system just discussed.

During UNIX installations, SSI_BASE must be set manually. The NT Install Shield, however, automates SSI_BASE

Understanding SSI_BASE environment variable

The SSI_BASE environment variable is an integral part of the FACTS security system, which verifies that the number of users signed onto the system do not exceed the number licenses purchased.

What SSI_BASE is set to and where it is set depends on how terminals — or PCs acting as terminals — connect to the UNIX host.

- *Dumb terminals with serial connections:* Use the primary tty as the SSI_BASE value, since the primary tty is a consistent and unique ID for each terminal/keyboard logging into FACTS. Set the SSI_BASE variable in each user's **.profile** on the UNIX host, rather than the FACTS7 startup script.

Example: **SSI_BASE=`tty`**
 export SSI_BASE

FacetTerm users need to make sure they set SSI_BASE *before* they start this program. Once the windowing package starts, it returns psuedo tty values, not primary values. Because psuedo values are different for each window, each would be counted as a separate user.

Placing SSI_BASE in **.profile** helps avoid this problem since UNIX refers to **.profile** before the FACT7 startup script where FacetTerm is launched.

- *PCs running terminal emulation using a serial connection:* This is essentially the same as a dumb terminal configuration, so treat it as such when setting SSI_BASE. Use the primary tty value and set the variable in the user's **.profile** on the UNIX host.
- *PCs on a network connection (TCP/IP or telnet) running **WindX** terminal emulation:* Set SSI_BASE to a unique value (such as the PC's IP address) and set it at the beginning of the **autoexec.bat**. WindX is the recommended terminal emulation solution for FACTS.

Other terminal emulation packages can be used on TCP/IP or telnet connections; however, they require a different configuration, and users may experience more tracking errors than normal.

If a non-supported emulation package is installed, SSI_BASE must equal a unique value, such as \$logname, and the variable needs to be set in the FACTS7 startup script. (Note to SCO users: Make sure \$LOGNAME is capitalized.) Administrators should also add a line to this script that says, in effect, set SSI_BASE if it isn't already set. Otherwise, the environment variable set in the FACTS7 startup script may overwrite the variables created in the **.profiles**.

Example: SSI_BASE=\${SSI_BASE:- "\$LOGNAME"}

Keep in mind, though, that the \$logname value restricts the way users log into the operating system. They cannot log in to multiple terminals with the same UNIX signon. If users *do* try to log on to FACTS while they are

PCs connected to a UNIX host don't have unique network IDs as they do in NT networks. Use each PC's IP address as the unique SSI_BASE value.

To find the IP address for each PC, right mouse click on Network Neighborhood and select the Configuration tab.

simultaneously logged into to different terminals, they may report user tracking errors or complain that the system “mysteriously” kicks them out to the FACTS banner screen for no reason.

- *Combination of serial and non-serial terminals:* If a UNIX system contains a combination of these terminal configurations, set SSI_BASE in all of the possible places using the appropriate values in each:
- in each user's .profile on the UNIX host to catch the dumb terminals and PCs emulating dumb terminals using serial connections (SSI_BASE=primary tty value).
- at the beginning of each autoexec.bat for PCs running WindX terminal emulation over a TCP/IP or telnet connection (SSI_BASE=PC network ID or some other unique value)
- in the FACTS7 startup script to catch PCs running unsupported terminal emulation software (SSI_BASE=\$logname).

Quick Guide to Setting SSI_BASE Variables

On	set SSI_BASE=	Set it where	Dependencies
UNIX systems running			
Dumb terminals	Primary `tty` value	Each user's .profile on UNIX host	Set SSI_BASE before starting FacetTerm. FT causes pseudo tty values
PCs emulating dumb terminals using serial connection	Primary `tty` value	Each user's .profile on UNIX host	Same as above
PCs running WindX terminal emulation	PC network ID, or any unique value (up to 20 characters long)	The beginning of the autoexec.bat file for each PC logging into FACTS	
PCs running other terminal emulation package connecting via TCP/IP or telnet	\$LOGNAME	FACTS startup script	Users must sign on from same location each time. Higher likelihood of user tracking errors. This configuration is not recommended.
Combination of all four terminal types	See above for UNIX systems	All of the above	PCs running terminal emulation software other than WindX may cause user-tracking errors.
NT systems			
	PC network ID, or any unique value (up to 20 characters long)	The beginning of the autoexec.bat file for each PC logging into FACTS	

Understanding WindX and the pvxhost script

WindX can help improve performance on FACTS systems running on ProvideX by offering a thin-client solution for PCs operating in NT and UNIX environments.

The program is designed to run on a direct TCP/IP connection. TCP/IP is not the only connectivity solution for WindX clients, however.

The software also runs on *serial connections*, but users lose windowing capabilities. Telnet is another option that offers greater security, but it may compromise speed.

To achieve maximum performance, the majority of WindX installations use direct TCP/IP connection by way of two programs – *NTHOST and *NTSLAVE – to provide communications between WindX clients and hosts running ProvideX.

*NTHOST and *NTSLAVE

*NTHOST is a server-side program that runs continuously in the background. It exists solely to monitor a particular TCP socket number – 10000 by default – and respond to calls from clients.

*NTSLAVE, on the other hand, is the client-side program that retains the server's TCP/IP address and the socket number the server monitors. It also tells ProvideX which program to run so that the FACTS Banner Screen is the first thing users see after they double-click the WindX desktop icon.

The process begins when a user double clicks on the WindX desktop icon and *NTSLAVE calls the server at the TCP/IP address and socket number provided.

When *NTHOST hears this call, it calculates the next open socket number on the server and spawns a new ProvideX session running the program indicated on the WindX command line, usually SSIWDX. It then instructs the WindX client to renegotiate a connection on a new socket where a new ProvideX session is running.

Pvxhost script

The FACTS install script places *pvxhost* in the **ssi7** directory. This script runs *NTHOST on the server. During UNIX installations, a line must be added to end of **etc/inittab** to respawn *NTHOST.

The following is an example of what the pvxhost script looks like after installation.

```
#!/bin/sh
PVXLIB=/usr1/ssi7/pvx/lib;export PVXLIB
```

In NT systems, there is no *respawn* equivalent. During the NT server installation creates an NT HOST shortcut icon on the desktop.

If you need to restart the server, double click this icon to restart the NT Host script.

```
FHOMEDIR=/usr1/ssi7
PVXDIR=${FHOMEDIR}/pvx
MODS_DIR=${MODS_DIR:-"${FHOMEDIR}/mods"}
RUNPROG=${FHOMEDIR}/prog/SM/SME100

SSI_USE_FACET=${SSI_USE_FACET:-"N"}
# "Y" - Start FacetTerm when starting FACTS
# "N" - Do not start FacetTerm when starting FACTS

WINDOWS=${WINDOWS:-"2"}
# Number of FacetTerm windows that will be used (if SSI_USE_FACET = "Y")

SSI_NO_FACET=${SSI_NO_FACET:-"1"}
# "0" - Update FacetTerm menu with current FACTS program from menu driver
# "1" - Do not update FacetTerm menu (speeds up FACTS menu driver)

SSI_NO_OA=${SSI_NO_OA:-"N"}
# "N" - Check for unread OA messages in FACTS menu driver
# "Y" - Do not check OA for unread messages (speeds up FACTS menu driver)

UMSK=0
MEM=512

export FHOMEDIR
export MODS_DIR
export SSI_USE_FACET
export WINDOWS
export SSI_NO_FACET
export SSI_NO_OA

umask ${UMSK}
cd /usr1/ssi7
/ssi7/pvx/pvx \*nthost -arg 10000 FACTS7_user 000 10999
</dev/null >/dev/null 2>/dev/null
```

Modifying pvxhost (UNIX installs only)

Notice that the last line contains the *NTHOST arguments.

10000 represents the **tcp_socket** number that the server monitors. *NTHOST defaults to 10000, but it is possible that this port is already assigned to another service. Check /etc/services to make sure socket 10000 is available and it is followed by a block of open sockets so that *NTHOST has room to assign new socket numbers. If not, find a socket that meets these requirements, but make sure the socket is greater than 5000. Anything less than the 5000 block is reserved for Internet services.

FACTS_user indicates the **user_id**. ProvideX switches to this user (*su*) when it starts new sessions. **Therefore, any permissions assigned to this user ID will be passed on to all WindX clients.** Do not assign **root** and make sure the user ID does not require a password.

000 is the **umask**.

10999 is the **max_port**. This argument is optional, but when it's used it tells *NTHOST the maximum port it is allowed to assign. Administrators may need to set this to keep WindX from interfering with other services.

</dev/null {space}>/dev/null {space}2>/dev/null prevents the activation key from becoming corrupt when *NTHOST is running in the background from /etc/inittab.

➤ In the beginning of the *NTHOST script, the *FacetTerm* environment variable is switched off. *This must be done* or *FacetTerm* will undermine client-server communications between WindX and ProvideX.



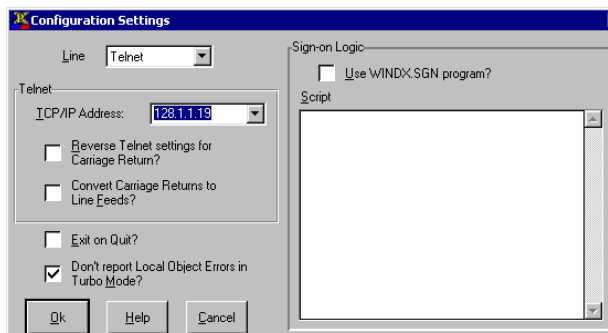
Technical Insight: Alternative WindX configurations

Telnet as an option

In general, the *NTHOST/*NTSLAVE solution provides the best performance for WindX users, but Telnet is another alternative.

In a Telnet configuration, WindX acts as a smart terminal emulator, logging into the UNIX server as if it were a terminal. But once ProvideX starts, WindX switches into thin-client mode, providing users with all the graphical functionality they would expect on a PC.

Because Telnet forces users to go through a UNIX prompt, it



provides more security than the straight TCP/IP approach. However, according to ProvideX, transmission speeds may degrade to as much as a tenth of a direct client-server connection.

Configuration involves fewer steps, since Telnet connections don't use *NTHOST and all the setup happens client-side:

1. Start WindX and select *Settings* from the top menu bar. (The *Settings* option may not appear if you start WindX from a shortcut. Try opening WindX from the Start menu, i.e., *Start*→*Programs*→*WindX*).
2. In the Configuration Settings window, set Line to Telnet. Then enter the server's TCP/IP address and click OK.
3. Choose **Connect**.

Users should see a standard UNIX sign on prompt, at which they can enter their sign on and password. Once WindX sees the ProvideX session, it will switch into thin-client mode and users can operate in the FACTS graphical user interface.

After their initial sign on, users simply need to double-click the WindX icon, click **Connect** and enter their sign on and password.

Using Multiple *NTHOST sessions on UNIX

It is possible to run multiple *NTHOST sessions on a UNIX system so that several different groups can connect to the host – each signing on with different permissions. This option gives administrators more flexibility in the area of security. To do this:

Create two *NTHOST scripts. In the *NTHOST task of the first script, set

- a socket number
- a user ID with one set of permission levels
- and a maximum port so the first session doesn't step on the second session.

Then in the second script, set

- a *different* socket number
- the user ID with different set of permissions
- and if needed, a new maximum port.

For example, the first script might end with

```
/ssi7/pvx/pvx \*nthost -arg 10000 Jim 000 12500 </dev/null >/dev/null 2>/dev/null
```

And at the end of the second script, the following might appear

```
/ssi7/pvx/pvx \*nthost -arg 13000 Richard 000 14500 </dev/null >/dev/null 2>/dev/null
```

Notice that the socket numbers are spread out over a large enough block to give *NTHOST room to assign socket numbers as clients connect.

2. Run both scripts from /etc/inittab to complete the server-side configuration.

3. Configure the clients by creating WindX desktop shortcuts for each client and, in the **Target**, enter the socket number in the *NTSLAVE statement that corresponds to the appropriate user ID with the appropriate permission level.

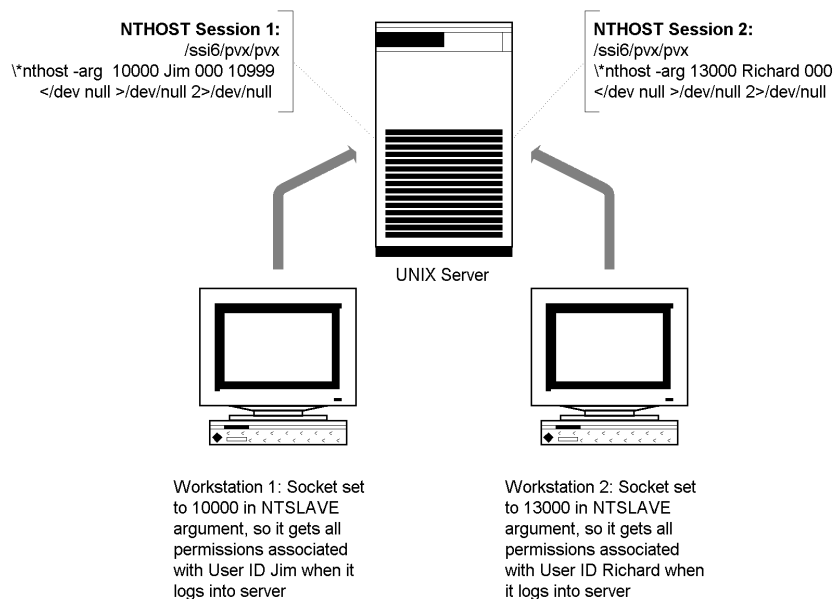
For example, clients that need server permissions associated with user_id Jim would see the following in the Target line of their WindX shortcuts

```
c:\pvx\pvxwin32.exe *ntslave -arg 151.2.11.34 SSIWDX 10000
```

Clients requiring server permissions associated with user_id Richard would see this Target line in their WindX shortcuts

```
c:\pvx\pvxwin32.exe *ntslave -arg 151.2.11.34 SSIWDX 13000
```

Multiple NTHOST Sessions



Additional resources and documentation

The Installation CD contains installation and user documentation for the following third-party products:

- FacetTerm (30-day trial installation)
- FacetWin (30-day trial installation)
- ICVERIFY
- General Report Writer
- Uniform
- VSI-FAX Gold Series Server and Client

If you need additional information on these products, or you need installation documentation for other third-party products, review the following contact information.

For additional information on...

ProvideX, WindX and ODBC

Clippership and TracerX (user doc)

Electronic Data Interchange (EDI)

Archive (Designed Data Systems, Inc.)

VSI-FAX (V-Systems, Inc.)

FacetTerm and FacetWin

General Report Writer and Uniform

Adobe Acrobat Reader and PDF files

Go to...

www.pvx.com

www.kewill.com

sfisher@dynitech.com or amy@scorpsys.com

(336) 272-1241

www.vsifax.com

www.facetcorp.com

www.synergetic-data.com

www.adobe.com