TN3270 Plus User Guide

TN3270 Plus User Guide

Printed: September 2025

Table of Contents

1 Overview

1.1 Introduction

TN3270 Plus is a flexible, efficient and inexpensive client application for connecting Windows PC users to IBM zSeries (mainframe), iSeries (AS/400) and UNIX systems via TCP/IP. TN3270 Plus includes terminal emulation for 3270, 5250, VT100, VT220 and ANSI terminals plus an integrated set of TCP/IP utilities in a compact easy to use product.

TN3270 Plus supports Windows 11, 10, 8, 7, XP and Windows Server 2022, 2019, 2016, 2012, 2008, 2003 and 2000. A common user interface to these operating systems allows deployment of the product throughout your enterprise without the support costs associated with multiple user interfaces. Tailor the desktop interface to your specifications with keyboard mapping, color definition and customizable ASCII to EBCDIC translation tables.

TN3270 Plus has many of the features of large expensive products in a tight efficient package for outstanding performance with minimal resource usage. For example, automate common tasks with the scripting language or connect up to 99 terminal emulation sessions of any type in any combination at the same time. Use the Internet utilities to gather information about other Internet users or to check the status of your e-mail without ever leaving TN3270 Plus. Why pay more for a product overloaded with features you will never use.

See Also:

Working with Multiple Sessions

1.2 System Requirements

TN3270 Plus comes in a 32-bit version that runs on Microsoft Windows 11, 10 (32-bit and 64-bit), 8 (32-bit and 64-bit), 7 (32-bit and 64-bit), XP (32-bit and 64-bit) and Windows Server 2022, 2019, 2012, 2008 (32-bit and 64-bit), 2003 (32-bit and 64-bit), and 2000. To run TN3270 Plus you need the following:

32-Bit Version

- Microsoft Windows 11, 10 , 8, 7, XP or Windows Server 2022, 2019, 2016, 2012, 2008, 2003 or 2000.
- 10 MB of available hard disk space.
- SVGA (800 x 600), 16-color or higher resolution monitor.
- TCP/IP.
- A connection to the Internet or an intranet.

1.3 Restrictions

Extended Attributes

The following 3279 EAB features are not supported by TN3270 Plus:

1. Field validation
2. Field outlining
3. Loadable character sets
4. Transparency

Keyboard Mapping

Due to Windows restrictions, the following key combinations cannot be mapped with TN3270 Plus keyboard mapping.

- Alt+F6
- Alt+Esc
- Alt+Tab

- Ctrl+Esc
- Ctrl+Alt+Del

VT100/VT220 Terminal Session Support

TN3270 Plus VT100 and VT220 terminal emulation sessions do not support DEC protected areas.

Office Vision

TN3270 Plus 5250 terminal emulation sessions do not support IBM's Office Vision.

3270 and 5250 Terminal Session Support

TN3270 Plus 3270 and 5250 terminal sessions do not support GDDM graphics.

3270 and 5250 Printer Session Support

TN3270 Plus 3270 and 5250 printer sessions do not support color printing.

1.4 Evaluation and License Agreements

Evaluation

TN3270 Plus is distributed as an evaluation version for a 30 day trial. The evaluation version includes all the product features. After the 30-day evaluation period, you must register the product with SDI if you wish to continue using the product. When you register the product, you also select the features you desire. See our How to Order web page for information on how to order TN3270 Plus.

License Agreement

By registering for and/or ordering TN3270 Plus, by whatever means, you agree to be bound by the terms and conditions of the Licensing Agreement (EULA).

1.5 Pricing

The per copy registration prices are found on the following web page:

TN3270 Plus Prices

Please refer to the Licence Agreement and the Maintenance and Technical Support Agreements for SDI's exclusive terms and conditions of use and sale.

See the our How to Order web page for information on how to order TN3270 Plus.

1.6 Registration Instructions

An evaluation version of TN3270 Plus may be downloaded from the TN3270 Plus download web page.

The evaluation version of TN3270 Plus will run for 30 days. After 30 days, TN3270 Plus will no longer operate. To convert your evaluation version into a licensed version you must register the product with SDI and receive a license code.

Registering TN3270 Plus

Instructions for ordering TN3270 Plus can be found on our How to Order web page.

After you complete your order, a license code will be e-mailed to you.

1.7 Registration Form

This form is no longer used. Please visit our How to Order web page to see instructions for ordering TN3270 Plus.

1.8 Copyright and Legal Notices

TN3270 Plus SSL utilizes the "OpenSSL toolkit" functionality provided by "The Open SSL Project" at http://www.openssl.org. SDI Limited acknowledges all patent rights therein."

The OpenSSL toolkit is licensed under a dual-license (the OpenSSL license and the original SSLeay license). See the license text below.

OpenSSL License

Copyright (c) 1998-2000 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org)
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

SSLeay license

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscape's SSL. This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)". The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.

1.9 Support/Questions/Suggestions

The best way to contact SDI about support or with questions or suggestions is to send an e-mail message to support@sdisw.com. You may also write or call:

SDI USA, Inc.
PO Box 5801
San Mateo, CA 94402
650 572 1200

The web site contains an FAQ (Frequently Asked Questions) web page with answers to commonly asked questions. Please review this page before contacting SDI.

Technical support is available at no charge during product evaluation and for 30 days after product registration. After the initial 30 day period, technical support is free to users paying annual maintenance and chargeable to other users. A complete description of the product support agreement can be found in the Maintenance and Technical Support Agreements section.

Please include the following information with any support request:

1. What version of Windows you are running.
2. What version of TN3270 Plus you are running. Select **About TN3270 Plus...** from the **Help** menu to get the TN3270 Plus version number and build date.
3. What terminal type are you emulating?
4. Can the problem be recreated? If so, how?
5. Include the full text of any dialog or message boxes that appear when the problem occurs.
6. Your telephone number and e-mail address.

SDI welcomes any questions about TN3270 Plus and we will do our best to answer those questions in a timely manner.

SDI would also like your input about possible enhancements to TN3270 Plus. We are interested in anything that will make TN3270 Plus easier to use, more efficient or more effective. Many of the product's features came from suggestions from users of the product.

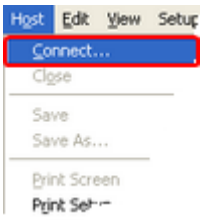
2 Getting Started

2.1 Connecting for the First Time

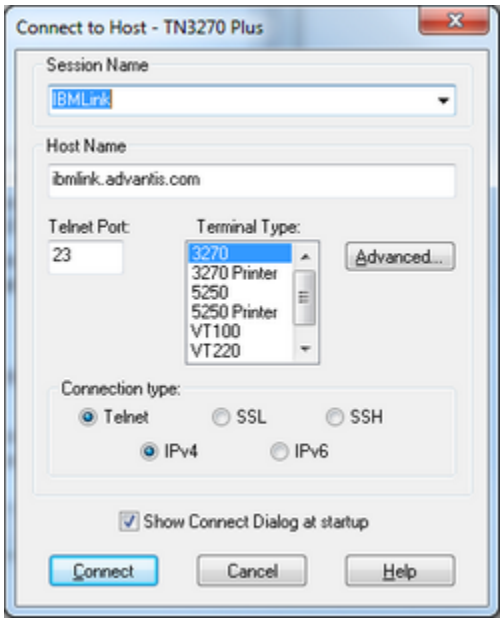
To get started with TN3270 Plus, you need to create a session and connect to a host computer. Once you are connected to the host computer, TN3270 Plus looks and acts like a local terminal cabled to the host computer.

Step-by-Step

1. If the **Connect to Host** dialog box is already displayed skip to step 2, otherwise open the **Host** menu and select the **Connect...** command or click the Connect button on the toolbar.



2. Fill in the Connect to Host dialog box. Here a few hints to get you started.



Session Name Enter a descriptive name for your session. This session will contain all the settings you need to connect to a specific host. If want to connect to more that one host computer, you will create multiple sessions. The

session name is used to select a tailored session for future connections. Here are some examples: (IBM, Mainframe Site 1, AS/400, UNIX, LINUX).

Host Name Enter the domain name or IP address of the computer you want to connect to. If you do not know the domain name or IP address, you will have to contact the network administrator at the host computer site. SDI cannot provide you with that information.

Telnet Port Enter the telnet connection port number. This is usually 23.

Terminal Type Select a terminal or printer type for your connection. You must select the correct type or your connection will not work correctly or may not work at all.

3270	Used for connections to an IBM zSeries or S/390 mainframe computers.
5250	Used for connections to an IBM iSeries or AS/400 midrange computers.
VT100	Used for connections to UNIX systems.
VT220	Used for connections to UNIX systems. VT220 is VT100 plus some additional functionality.
ANSI	Used for connections to UNIX systems.

The **3270 Printer** and **5250 Printer** items are for printer sessions, see Connecting a TN3287 Printer session and Connecting a 5250 Printer Session for more information on printer sessions.

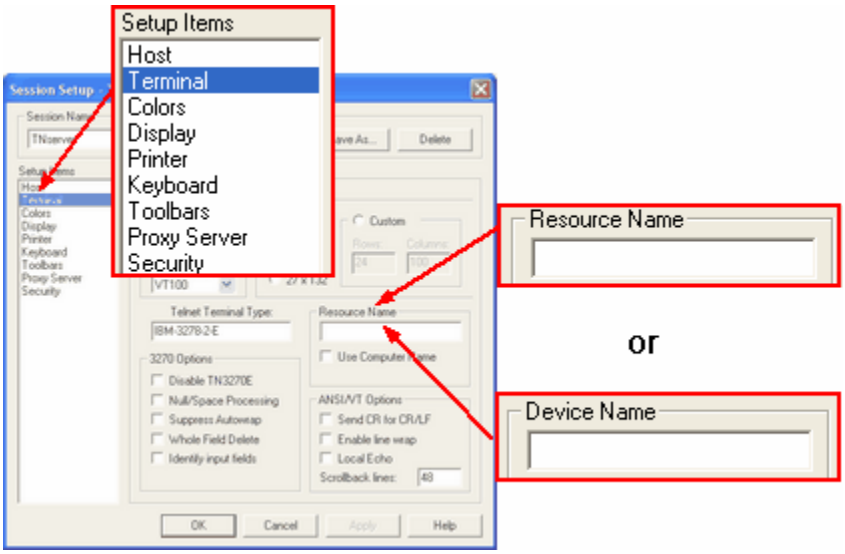
Connection Type Select the connection type.

Telnet	No encryption.
TLSv1	TLS (Transport Layer Security) version 1.0 encryption.
SSH	SSH (Secure Shell) encryption.

IPv4	IPv4 connections.
IPv6	IPv6 connections.

3. This is all you need for connecting to most telnet servers. However, if you are connecting to an IBM zSeries (mainframe) or iSeries (midrange) computer (3270 or 5250 terminal type) and were given an LU name or Device name you need to add that to your session settings. (If you were not given an LU name or Device name you can skip this step.)

   A. Click the **Advanced...** button to display the **Session Setup** dialog box.
   B. In the **Session Setup** dialog box, select **Terminal** in the **Setup Items** list.
   C. Fill in the Resource Name or Device Name edit box and click the OK button.



4. Click the **Connect** button to save your session and make the connection.
5. The following links contain to the solutions for common connection problems.

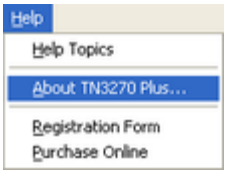
Failed to Create New Socket
Host www.xxx.yyy.zzz Is Unreachable
Host www.xxx.yyy.zzz Is Unreachable: Socket error 0: WSAEUNDEFINED
Host www.xxx.yyy.zzz Is Unreachable: Socket error 10022: WSAEINVAL
Socket Error 10060: WSAETIMEDOUT
Socket Error 10061: WSAECONNREFUSED
Disconnected by host www.xxx.yyy.zzz

2.2 Entering Your License Code

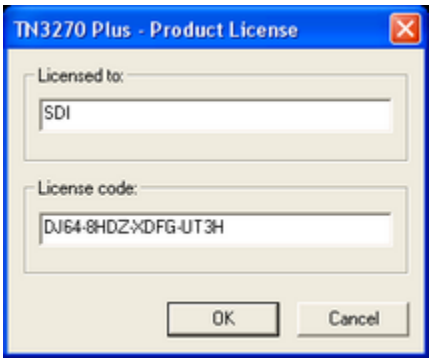
TN3270 Plus is distributed as an evaluation version for a 30 day trial. The evaluation version includes all the product features. If you want to continue to run TN3270 Plus after the 30-day evaluation period has expired, you must purchase a license from SDI. See our How to Order web page for information on how to purchase a license for TN3270 Plus.

When you purchase a license, you will receive a license code from SDI. To enter the license code:

1. Start TN3270 Plus.
2. Open the **Help** menu and click on **About** TN3270 Plus**...**



3. If the **About TN3270 Plus Evaluation Version** dialog box appears, click on the **License Code** button. If the **About TN3270 Plus** dialog box appears, click on the **License...** button.
4. In the **Product License** dialog box, fill in the **Licensed to** and **License code** edit boxes. Both the **Licensed to** and **License code** fields are case sensitive. A portion of the licensed to name is encrypted into the license code, so both fields must be entered correctly for the license code to work. Click the **OK** button.



When the TN3270 Plus license code is entered, it is stored in the tn3270.ini file. The default location for this file is:

Operating System	tn3270.ini location
2022 / 2019 / 2016 / 2012 / 2008 / 11/ 10 / 8 / 7	C:\ProgramData\SDI\TN3270 Plus\
2003 / XP / 2000	C:\:\Documents and Settings\All Users\Application Data\SDI\TN3270 Plus\

If you wish to store the license code in a different location, see Moving the License Code File.

Error Messages:

The license code is not valid for this release

The license code you entered is for an older release of TN3270 Plus.

The license code is generated for a specific release and will not function with newer releases of TN3270 Plus.

The product release level is three digits separated by periods, for example, 3.3.0. The license code only checks the first two digits, so a license code created for 3.2.0 will work with releases 3.2.1 through 3.2.9. However, if you try to use it with release 3.3.0 or higher you will get the above error message.

If you have subscribed to our maintenance and support agreement, updates are free. Send an e-mail to sales@sdisw.com with your current "licensed to" name and the new TN3270 Plus release level and you will be e-mailed a license code for the new release.

The 'Licensed to' file is incorrect

The name in the "Licensed to" name does not match the name encrypted into the license code. The "Licensed to" name is incorrect or contains a typographical error.

1. The "Licensed to" name is case sensitive make sure all letters are in the correct case.
2. Make sure the "Licensed to" name has the correct number of spaces.
3. The "Licensed to" name and license code are issued in pairs. The "Licensed to" name must correspond to the license code you received.

The 'License code' field is incorrect

The "License code" field contains a typographical error. Make sure the license code is all in upper case and that all the characters are correct.

2.3 Printing with TN3270 Plus

Printing with TN3270 Plus sometimes causes some confusion because there are four different ways to print.

1. Print screen.
2. Printer session
3. SDI LPD
4. Printer passthrough

Summary

The following table summarizes the differences between the printing methods. You can find more detail about each method following the table.

Feature	Licence Requirements	Supported Terminal Types	Supported Host Computers
Print screen	None	All	All
Printer session	3270 Terminal Feature or 5250 Terminal Feature Print Feature	3270, 5250	IBM zSeries (mainframe) IBM iSeries (AS/400)
SDI LPD	Print Feature	All	All (LPR is required on the host computer)
Printer passthrough	None	VT100, VT220, ANSI	UNIX

Print Screen

"Print Screen" support is just what it sounds like. It lets you print the contents of the current screen. You can print a single screen on a local or network printer. This is a standard feature and works for all terminal types.

If you want to print multiple screens you can turn on TN3270 Plus logging and save screens to a log file. After you have logged all the screens you want to save you can turn logging off and print the log file.

Printer Session

A printer session allows you to send host computer print jobs via to TN3270 Plus to a Windows printer. A printer session looks like a printer to the host computer. Your printer session can print jobs on any local or remote Windows printer.

Printer sessions are supported only by IBM zSeries (mainframe) and iSeries (AS/400) host computers. You cannot create a printer session for a UNIX host computer. You must license the TN3270 Plus **Printer Support.** You must also have the corresponding terminal feature, so the **3270 Terminal** and/or the **5250 Terminal** feature is also required.

Printer definitions must be properly configured on the host computer before you can use printer sessions. If you want to use a printer session, you need to get a printer LU name (IBM zSeries) or a printer device name (IBM iSeries) from the network administrator for the host computer.

See Connecting a TN3287 Printer Session and Connecting a 5250 Printer Session for more information.

SDI LPD

SDI LPD is a Line Printer Daemon and it also allows you to send host computer print jobs to a Windows printer. SDI LPD is a separate executable program (sdilpd.exe). While it is running it awaits print jobs from a Line Print Requester (LPR). You can find out more about how SDI LPD works in the . You must license the TN3270 Plus Print feature to enable SDI LPD.

Since Printer Sessions and SDI LPD accomplish the same goal, how do you choose which you need to use? The biggest advantage of printing with SDI LPD is that almost all host computers support LPR and can send print jobs to SDI LPD. This includes UNIX systems which do not support printer sessions. The biggest disadvantage is that the host computer must send the print job to the IP address of the PC running SDI LPD. This becomes a problem if you connect to a host computer via an Internet dial-up connection because dial-up connections are assigned a new IP address every time they connect. This means the LPR commands on the host computer must be changed to send print jobs to your new IP address each time you make a new dial-up connection to the Internet. Printer sessions use the LU name or Device name to identify the printer session and that name is associated with the IP address when the connection is made to the host computer. So, it does not matter if your IP address changes each time you connect to the Internet.

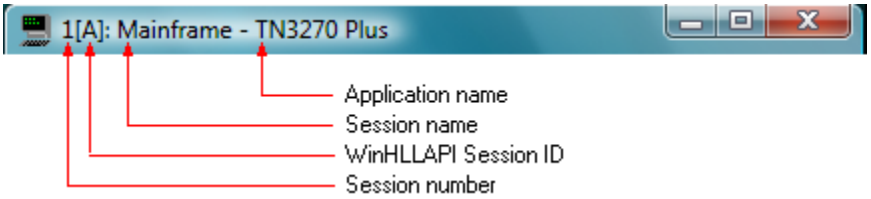
Printer Passthrough

Printer passthrough is a UNIX feature that sends data to a terminal printer. UNIX initiates printer passthrough by sending a control sequence to the terminal to start printer passthrough. Host data is sent to the printer until UNIX sends a control sequence to turn printer passthrough off. TN3270 Plus supports printer passthrough and sends the printer passthrough data to the Windows printer defined in the printer pane of the session setup dialog box.

3 How to...

3.1 Connecting to a Host Computer

1. Open the **Host** menu and select the **Connect** command or use the blue up arrow on the toolbar.
2. Fill in the Connect to Host dialog box and click the **Connect** button.
3. TN3270 Plus connects to the specified host. The session number is displayed on the application title bar. For example:



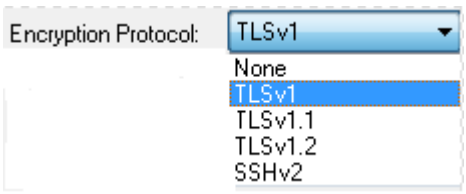
Tips:

- You may connect up to 99 sessions of any type in any combination in a single window. You may also start sessions in multiple windows. See Working with Multiple Sessions for more information.

3.2 Connecting to a Host Computer Using TLS

TLS (Transport Layer Security) and SSL (Secure Socket Layer) support allow you to make a secure connection to a host computer. The host computer must be configured to support secure connections.

1. Open the **Host** menu and select the **Connect** command or use the blue up arrow on the toolbar.
2. Fill in the Connect to Host dialog box. Make sure you specify the TLSv1 radio button. The TLSv1 radio button selection defaults to the TLS version 1.0 protocol with no server certificate options. If you need to modify the default configuration go to step 3 otherwise skip to step 8.
3. Click the **Advanced…** button and select **Security** under **Setup Items** to display the Security pane of Session Setup dialog box.
4. Select the desired TLS or SSL encryption protocol.



5. Specify the **Server Certificate** options. In most cases, the "Use Windows certificate store" is the only option you will need. This option imports the "Trusted Root Certification Authorities" certificates from the Internet Explorer certificate store into the TN3270 Plus certificate store (tn3270.pem). If the server

certificate is not signed by a Trusted Root Certification Authority (not common), place a copy of the entire server certificate chain on your PC. Enter the the full path and name of the server certificate chain file in the **Server Certificate File (.pem)** edit box. The server certificate chain must be in the proper order starting with the server certificate and ending with the root Certificate Authority certificate. The certificate(s) must be in ".pem" format. You can use notepad merge multiple certificates into a single file. See the Session Setup (Secuirty Pane) dialog for a complete description of each of the Server certificate options.

6. Specify the **Client Certificate** options. If the host computer does TLS client validation (not common), enter the full path and name of the client certificate file in the **Client Certificate File (.pem)** edit box. The certificate must be in ".pem" format. Specify the client certificate encryption password in the **Password** edit box.
7. Click the **OK** button to save your configuration options and close the dialog box.
8. Click the **Connect** button in the Connect to Host dialog box and TN3270 Plus creates a secure connection to the specified host.

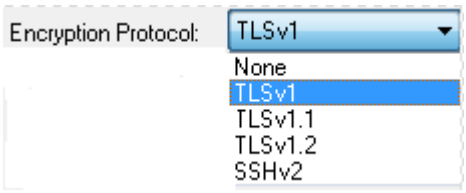
Tips:

- SSL support is an optional additional cost feature. If your copy of TN3270 Plus does not have the SSL feature, please contact sales@sdisw.com
- If you are making a secure connection to a host computer that is behind a proxy server, you may need to set up your session for "no SSL" to get connected to the proxy server. Then use the SSLConnect script command to create the secure connection to the host computer.
- You may connect up to 99 sessions of any type in any combination in a single window. You may also start sessions in multiple windows. See Working with Multiple Sessions for more information.
- TN3270 Plus SSL support includes software developed by the OpenSSL project for the OpenSSL toolkit. This feature includes cryptographic software written by Eric Young. This feature includes software written by Tim Hudson. (See the OpenSSL license agreement.)

3.3 Connecting to a Host Computer Using SSH

SSH (Secure Shell) support allows you to make a secure connection to a host computer. The host computer must be configured to support secure connections.

1. Open the **Host** menu and select the **Connect** command or use the blue up arrow on the toolbar.
2. Fill in the Connect to Host dialog box. Make sure you specify the SSH radio button. The SSH radio button selection defaults to the "SSHv1 or SSHv2" protocol password authentication. If you need to modify the default configuration go to step 3 otherwise skip to step 6.
3. Click the **Advanced…** button and select **Security** under **Setup Items** to display the Security pane of **Session Setup** dialog box.
4. Select the desired SSH encryption protocol.



5. Select the SSH configuration options. See the Session Setup (Secuirty Pane) dialog box for a description of each of the options. Click the **OK** button to save your configuration options and close the dialog box.
6. Click the **Connect** button in the Connect to Host dialog box and TN3270 Plus creates a secure connection to the specified host.

Tips:

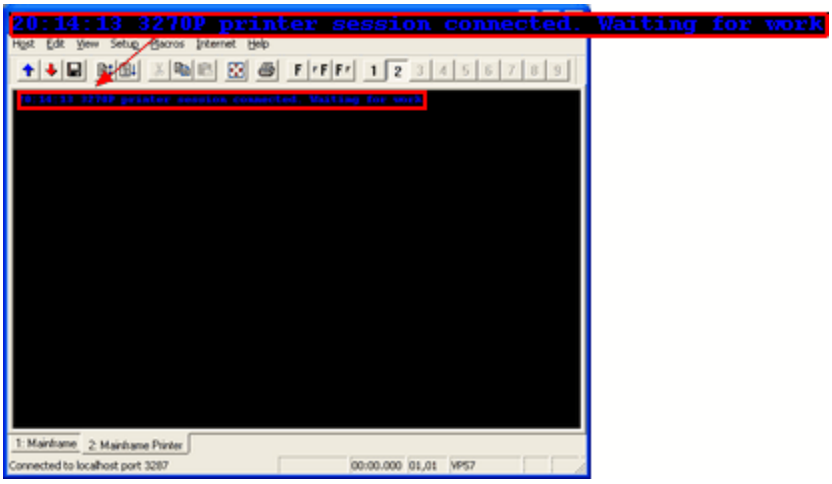
- SSH support is an optional additional cost feature. If your copy of TN3270 Plus does not have the SSH feature, please contact sales@sdisw.com
- You may connect up to 99 sessions of any type in any combination in a single window. You may also

start sessions in multiple windows. See Working with Multiple Sessions for more information.

3.4 Connecting a TN3287 Printer Session

1. Open the **Host** menu and select the **Connect** command or use the blue up arrow on the toolbar.
2. Fill in the Connect to Host dialog box. Specify **3270 Printer** as the terminal type.
3. Click on the **Advanced…** button and select Terminal under Setup Items to display the Terminal pane of **Session Setup** dialog box. Fill in the resource name for your printer session in the Resource Name edit box.

   If you wish to use a printer associated with an active terminal session, specify the resource name of the terminal session and check the **TN3270E Associate** check box. The terminal session must be connected and it must be defined with an associated printer in the configuration files on the host computer.
4. Click the **OK** button to return to the Connect to Host dialog box.
5. Click the **Connect** button to connect the printer session. If connection is successful, the printer session displays a message similar to the following:

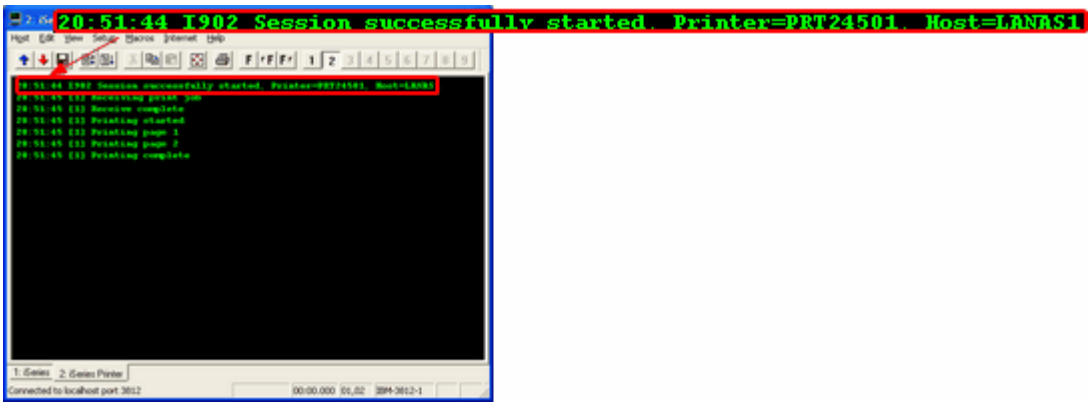


Tips:

- TN3287 Printer support is part of the TN3270 Plus Print feature. The Print feature is an optional additional cost feature. If your copy of TN3270 Plus does not have the Print feature, please contact sales@sdisw.com

3.5 Connecting a 5250 Printer Session

1. Open the **Host** menu and select the **Connect** command or use the blue up arrow on the toolbar.
2. Fill in the Connect to Host dialog box. Specify **5250 Printer** as the terminal type.
3. Click on the **Advanced…** button and select Terminal under Setup Items to display the Terminal pane of **Session Setup** dialog box. Fill in the device name for your printer session in the Device Name edit box.
4. Click the **OK** button to return to the Connect to Host dialog box.
5. Click the **Connect** button to connect the printer session. If connection is successful, the printer session displays a message similar to the following:



Tips:

- 5250 Printer support is part of the TN3270 Plus Print feature. The Print feature is an optional additional cost feature. If your copy of TN3270 Plus does not have the Print feature, please contact sales@sdisw.com

3.6 Changing Keypad Toolbar Button Text

You can assign your own text to the keypad toolbar buttons. To do this edit the keypad toolbar configuration file (for example, "3270 Default.ttb" for the 3270 default toolbar) and add text to the entries you want to change. For example, to change the text on the PF3 button from "PF3" to 'Help," change the PF3 entry from:

Button_nn=PF3

to:

Button_nn=PF3,Help

3.7 Creating a Backup Copy of TN3270 Plus

You should create a backup copy of TN3270 Plus to keep in case you need to reinstall due to a disk head crash or other problems. To create a backup, you should copy the following items to a removable media and store that media in a save place.

1. The TN3270 Plus distribution file. This is the file you used to install TN3270 Plus. (e.g. TN3270PlusSetup.exe or tn3270vm.exe where vm is the product version level.)

2. The TN3270 Plus license code information. The TN3270 Plus licenses code is kept in the TN3270.ini file. This file can be found in the TN3270 Plus configuration files folder (Setup, Preferences, Configuration File Folder edit box.) This is the default location. The default location can be overridden by either of the following registry entries:
   HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\LicenseCodePath\
   HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration\LicenseCodePath\ (64-bit Windows)
   HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\LicenseCodePath\ (32-bit Windows)

3. (Optional) The TN3270 Plus configuration files. The configuration files are kept in the TN3270 Plus configuration files folder. (Setup, Preferences, Configuration File Folder edit box.) If you do not backup these files, it will be necessary to recreate your configuration settings after TN3270 Plus is installed.

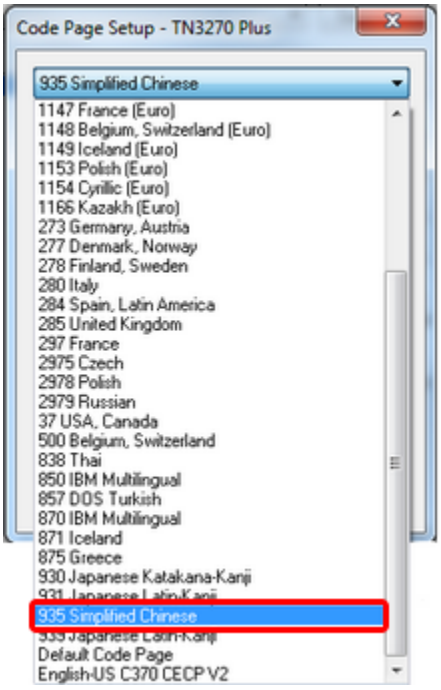
3.8 Entering Chinese Language Characters

TN3270 Plus (release 3.6.5 above) supports code pages for the Chinese language. This allows terminal emulation input and display of Chinese characters. Chinese is a DBCS (double-byte Character Set) language. In DBCS languages each character is consists of two bytes instead of one.

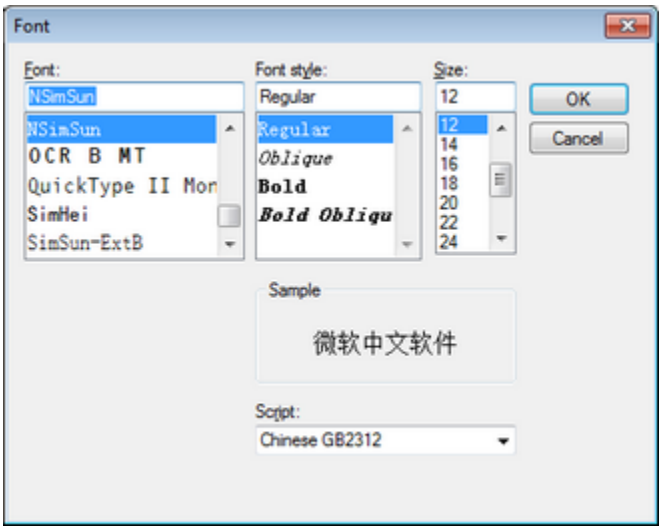
There a several steps that are required to display Chinese characters in TN3270 Plus.

1. Install Chinese language support in Windows. The Chinese language support installation process varies for different versions of Windows. Follow the installation process for your version of Windows.

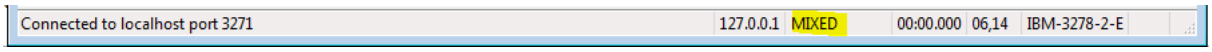
2. Specify a Chinese language code page for your TN370 Plus terminal session. To select a code page, open the **Setup** menu and select **Host...** to display the Host Pane of the **Session Setup** dialog box. In the **Session Setup** dialog box, click the **Browse...** button, that follows the Code Page edit box to display the Code Page Setup dialog box. In the **Code Page Setup** dialog box, click the down arrow at the right side of the drop-down list box and select a Chinese Language code page.
   .



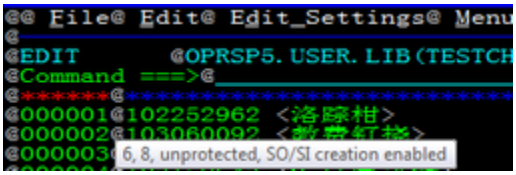
3. Select a TN3270 Plus display font that supports Chinese characters. To select a display font, open the **Setup** menu and select **Display...** to display the Display Pane of the **Session Setup** dialog box. In the **Session Setup** dialog box, click the **Change...** button, that follows the Screen Font edit box to display the Font dialog box. In the **Font** dialog box, select a Chinese font, a font style and a font size and select a Chinese script in the Script drop-down list box. "NSimSun" and "SimHei" are fonts that support Chinese characters.



4. The terminal emulation screen must contain fields that support DBCS (Double-byte Character Set) characters. When the cursor is in a field that supports DBCS characters you will see the word "MIXED" in the TN3270 Plus status bar.



An alternate way to determine if a field supports DBCS characters is to turn on the TN3270 Plus "Display attribute characters" option. (Setup, Display..., Display attribute characters check box) When this option is checked an at sign (@) is displayed in each screen position that contains an attribute byte. Attribute byte tips support is also activated. When this option is on, hovering the cursor over an attribute byte displays a small window containing a description of the attribute byte. "SO/SI creation enabled" will appear in the attribute byte description window for fields that support DBCS characters.



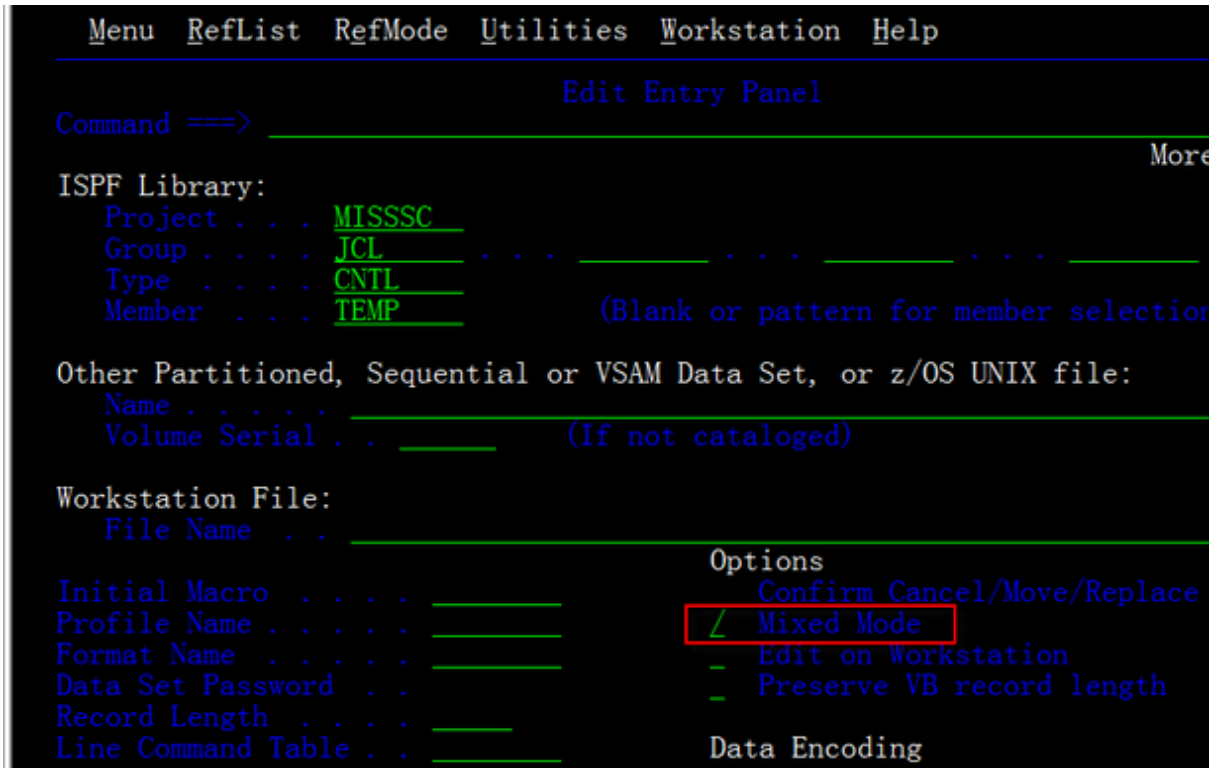
5. To type a Chinese character into a MIXED field you must select the Chinese language from the Windows Language bar.

Chinese characters take 2 single byte character spaces on the terminal emulation screen. When Chinese characters are entered they are enclosed in less than and greater than signs, < >. The less than sign indicates the start of Chinese characters in the field and the greater than sign indicates the end of Chinese characters in the field.

Enabling DBCS Character Entry and Display in ISPF

Specify the ISPF "Mixed Mode" option in the Edit Entry Panel and/or the View Entry Panel to enable you to view, browse or edit mixed data that contains both EBCDIC (single-byte) and DBCS (double-byte) characters in ISPF.

When DBCS characters are entered they are enclosed in SO ("<") and SI(">") characters.. The SO (less than sign) character indicates the start of DBCS characters in the field and the SI (greater than sign) character indicates the end of DBCS characters in the field.



Enabling DBCS Character Entry and Display in CICS BMS Maps

Specify the BMS map SOSI option to enable a field to contain both EBCDIC (single-byte) and DBCS (double-byte) characters . For example:

```
MAPATTS=(COLOR,HILIGHT,PS,SOSI)
```

When DBCS characters are entered into SOSI enabled fields they are enclosed in SO ("<") and SI(">") characters. The SO (less than sign) character indicates the start of DBCS characters in the field and the SI (greater than sign) character indicates the end of DBCS characters in the field.

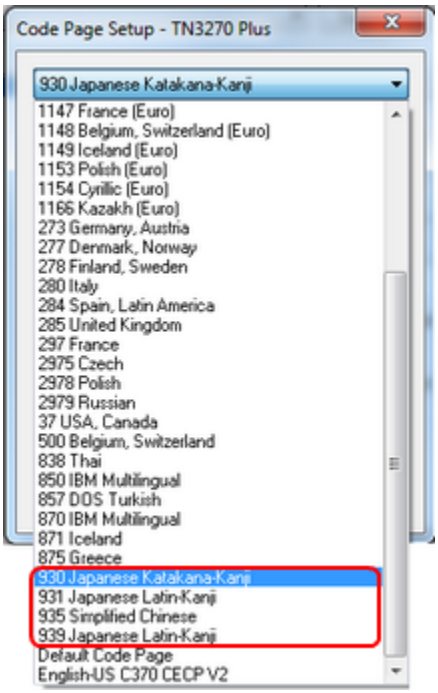
3.9 Entering Japanese Language Characters

TN3270 Plus (release 3.6.5 above) supports code pages for the Japanese language. This allows terminal emulation input and display of Japanese characters. Japanese is a DBCS (double-byte Character Set) language. In DBCS languages each character is consists of two bytes instead of one.

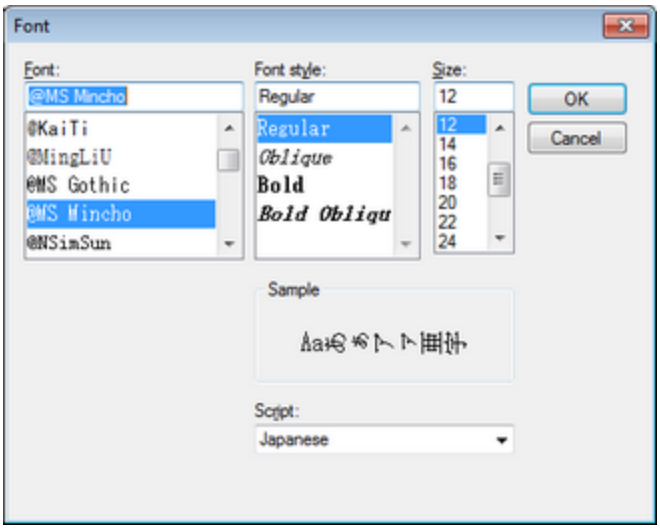
There a several steps that are required to displayJapanese characters in TN3270 Plus.

1. Install Japanese language support in Windows. The Japanese language support installation process varies for different versions of Windows. Follow the installation process for your version of Windows.

2. Specify a Japanese language code page for your terminal session. To select a a code page, open the **Setup** menu and select **Host...** to display the Host Pane of the **Session Setup** dialog box. In the **Session Setup** dialog box, click the **Browse...** button, that follows the Code Page edit box to display the Code Page Setup dialog box. In the **Code Page Setup** dialog box, click the down arrow at the right side of the drop-down list box and select a Japanese Language code page.

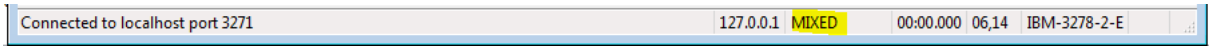


3. Select a TN3270 Plus display font that supports Japanese characters. To select a display font, open the **Setup** menu and select **Display...** to display the Display Pane of the **Session Setup** dialog box. In the **Session Setup** dialog box, click the **Change...** button, that follows the Screen Font edit box to display the Font dialog box. In the **Font** dialog box, select a Japanese font, a font style and a font size and select a Japanese script in the Script drop-down list box. "MS Mincho" and "MS Gothic" are fonts that support Japanese characters.



4.

5. The terminal emulation screen must contain fields that support DBCS (Double-byte Character Set) characters. When the cursor is in a field that supports DBCS characters you will see the word "MIXED" in the TN3270 Plus status bar.



An alternate way to determine if a field supports DBCS characters is to turn on the TN3270 Plus "Display attribute characters" option. (Setup, Display..., Display attribute characters check box) When this option is checked an at sign (@) is displayed in each screen position that contains an attribute byte. Attribute byte tips

support is also activated. When this option is on, hovering the cursor over an attribute byte displays a small window containing a description of the attribute byte. "SO/SI creation enabled" will appear in the attribute byte description window for fields that support DBCS characters.

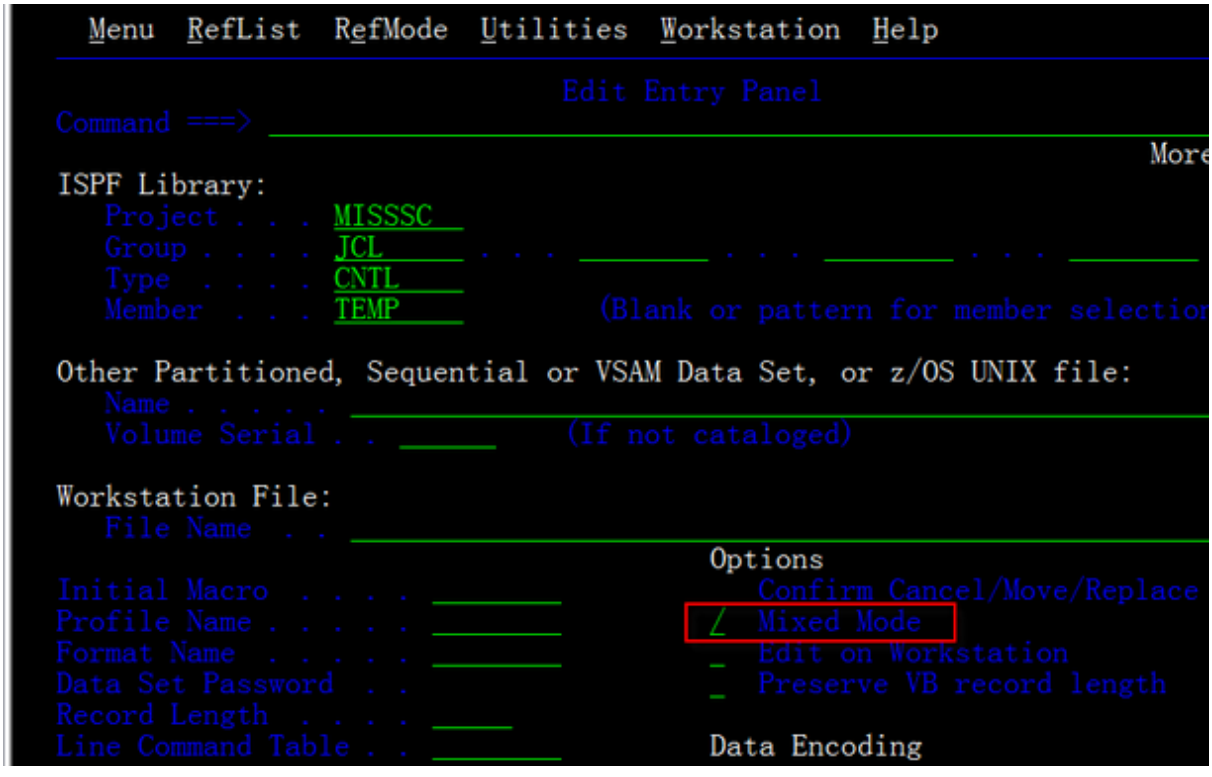
6. To type a Japanese character into a MIXED field you must select the Japanese language from the Windows Language bar.

Japanese characters take 2 single byte character spaces on the terminal emulation screen. When Japanese characters are entered they are enclosed in less than and greater than signs, < >. The less than sign indicates the start of Japanese characters in the field and the greater than sign indicates the end of Japanese characters in the field.

Enabling DBCS Character Entry and Display in ISPF

Specify the ISPF "Mixed Mode" option in the Edit Entry Panel and/or the View Entry Panel to enable you to view, browse or edit mixed data that contains both EBCDIC (single-byte) and DBCS (double-byte) characters in ISPF.

When DBCS characters are entered they are enclosed in SO ("<") and SI(">") characters.. The SO (less than sign) character indicates the start of DBCS characters in the field and the SI (greater than sign) character indicates the end of DBCS characters in the field.



Enabling DBCS Character Entry and Display in CICS BMS Maps

Specify the BMS map SOSI option to enable a field to contain both EBCDIC (single-byte) and DBCS

(double-byte) characters . For example:

```
MAPATTS=(COLOR,HILIGHT,PS,SOSI)
```

When DBCS characters are entered into SOSI enabled fields they are enclosed in SO ("<") and SI(">") characters. The SO (less than sign) character indicates the start of DBCS characters in the field and the SI (greater than sign) character indicates the end of DBCS characters in the field.

3.10 Entering Korean Language Characters

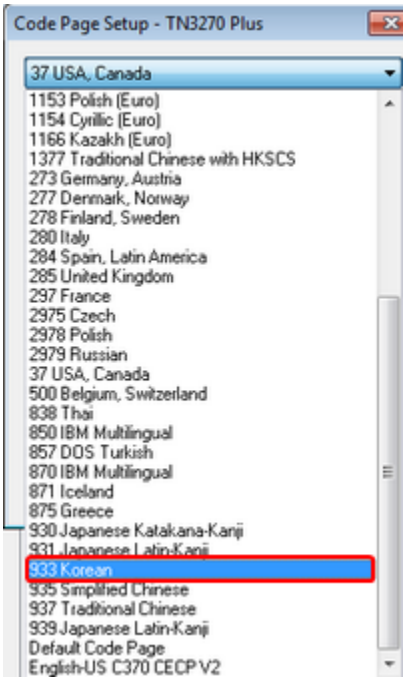
TN3270 Plus (release 4.0.5 above) supports a code page for the Korean language. This allows terminal emulation input and display of Korean characters. Korean is a DBCS (double-byte Character Set) language. In DBCS languages each character consists of two bytes instead of one.

There a several steps that are required to display Korean characters in TN3270 Plus.

1. Install Korean language support in Windows. The Korean language support installation process varies for different versions of Windows. Follow the installation process for your version of Windows.

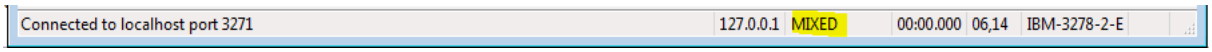
2. Specify the Korean language code page for your TN370 Plus terminal session. To select a code page, open the **Setup** menu and select **Host...** to display the Host Pane of the **Session Setup** dialog box. In the **Session Setup** dialog box, click the **Browse...** button, that follows the Code Page edit box to display the Code Page Setup dialog box. In the **Code Page Setup** dialog box, click the down arrow at the right side of the drop-down list box and select the Korean Language code page

   .



3. Select a TN3270 Plus display font that supports Korean characters. To select a display font, open the **Setup** menu and select **Display...** to display the Display Pane of the **Session Setup** dialog box. In the **Session Setup** dialog box, click the **Change...** button, that follows the Screen Font edit box to display the Font dialog box. In the **Font** dialog box, select a Korean font, a font style and a font size and select the Korean script in the Script drop-down list box.

4. The terminal emulation screen must contain fields that support DBCS (Double-byte Character Set) characters. When the cursor is in a field that supports DBCS characters you will see the word "MIXED" in the TN3270 Plus status bar.



An alternate way to determine if a field supports DBCS characters is to turn on the TN3270 Plus "Display attribute characters" option. (Setup, Display..., Display attribute characters check box) When this option is checked an at sign (@) is displayed in each screen position that contains an attribute byte. Attribute byte tips support is also activated. When this option is on, hovering the cursor over an attribute byte displays a small window containing a description of the attribute byte. "SO/SI creation enabled" will appear in the attribute byte description window for fields that support DBCS characters.

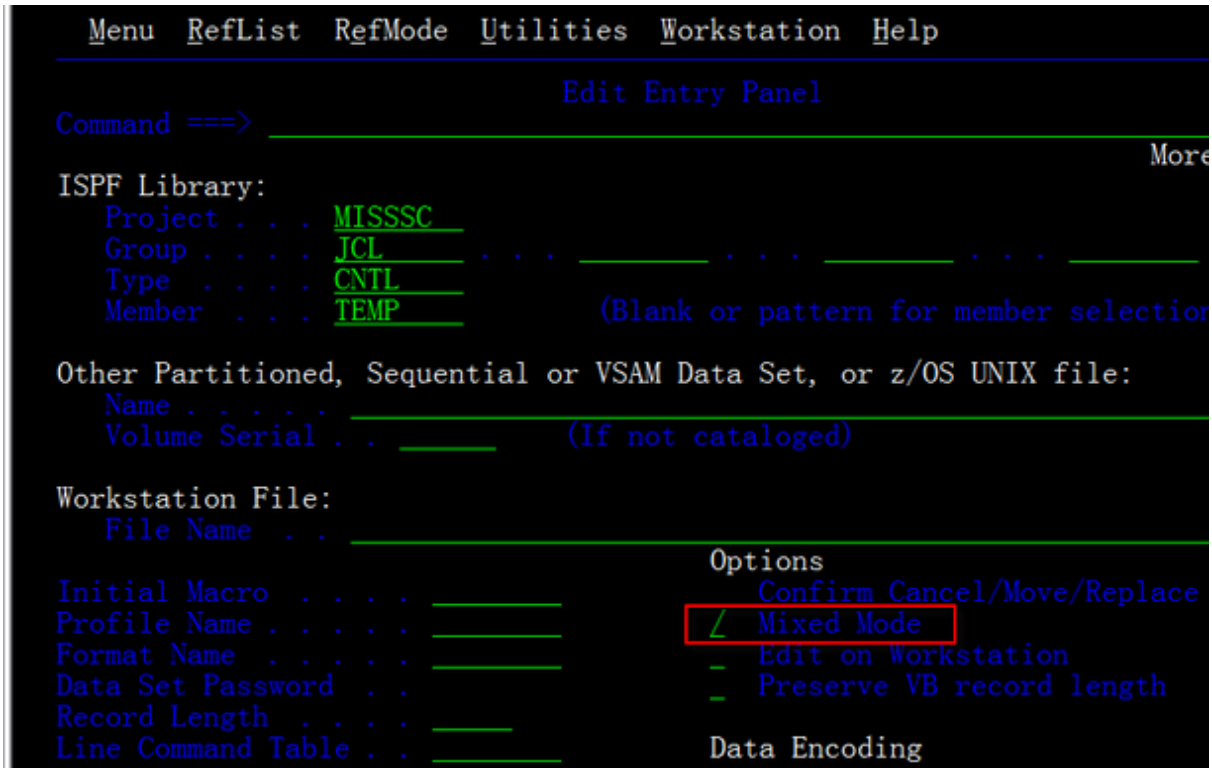
5. To type a Korean character into a MIXED field you must select the Korean language from the Windows Language bar.

Korean characters take 2 single byte character spaces on the terminal emulation screen. When Korean characters are entered they are enclosed in less than and greater than signs, < >. The less than sign indicates the start of Korean characters in the field and the greater than sign indicates the end of Korean characters in the field.

Enabling DBCS Character Entry and Display in ISPF

Specify the ISPF "Mixed Mode" option in the Edit Entry Panel and/or the View Entry Panel to enable you to view, browse or edit mixed data that contains both EBCDIC (single-byte) and DBCS (double-byte) characters in ISPF.

When DBCS characters are entered they are enclosed in SO ("<") and SI(">") characters.. The SO (less than sign) character indicates the start of DBCS characters in the field and the SI (greater than sign) character indicates the end of DBCS characters in the field.



Enabling DBCS Character Entry and Display in CICS BMS Maps

Specify the BMS map SOSI option to enable a field to contain both EBCDIC (single-byte) and DBCS (double-byte) characters . For example:

```
MAPATTS=(COLOR,HILIGHT,PS,SOSI)
```

When DBCS characters are entered into SOSI enabled fields they are enclosed in SO ("<") and SI(">") characters. The SO (less than sign) character indicates the start of DBCS characters in the field and the SI (greater than sign) character indicates the end of DBCS characters in the field.

3.11 Positioning the Cursor

To change the cursor position, press the arrow keys on the keyboard.

You may also position the cursor using the mouse. The default cursor movement setting is a single click of the left mouse button. You may change the cursor movement setting by updating the CursorMove terminal key function in the Keyboard Setup dialog box. (Setup, Keyboard, Configure… button, Function Group = Terminal Keys, CursorMove).

3.12 Specifying Parameters on the Command Line

TN3270 Plus supports the following command line switches. Use the command line switches to control what TN3270 Plus does at startup when you start it from a batch file or shortcut.

/Session ***session_name***	specifies the session to connect. Where session_name is the session name. If session_name contains embedded spaces it must be enclosed in double quotes.
/Host ***xx.xx.xx.xx[:port]***	is the IP address and optionally the port number of the host. If the port number is not specified it defaults to 23.
/Script ***script_name***	specifies a script to run after the session specified in "/Session" or /Host is connected. Where script_name is the name of the script. If script_name contains embedded spaces it must be enclosed in double quotes.

/Hidden	start TN3270 Plus in a hidden window. This option is useful if you are starting TN3270 Plus from a WinHLLAPI application and do not want users to be able to see the TN3270 Plus window.

Session Switch Example:

When the "/Session" switch is included, TN3270 Plus connects to the specified session when it starts.

Windows 10 (32-bit), 8 (32-bit), 7 (32-bit), XP and Windows Server 2008(32-bit), 2003 and 2000.

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" /Session "**P/390 27x132"**

Windows 11, 10 (64-bit), 8 (64-bit), Windows Server 2022, 2019, 2016, 2012, Server 2008 (64-bit), 7 (64-bit) and Vista (64-bit).

"C:\Program Files (x86)\SDI\TN3270 Plus\tn3270.exe" /Session "**P/390 27x132"**

Where:

P/390 27x132 is the session name.

Tips:

- The session name is case sensitive.
- If the session name contains any spaces, it must be enclosed in double quotes as in the example above.

Host Switch Example:

When "/Host" switch is included, TN3270 Plus connects to the specified IP address on the specified port number .

Windows 10 (32-bit), 8 (32-bit), 7 (32-bit), XP and Windows Server 2008(32-bit), 2003 and 2000.

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" /Host 192.168.1.5:23

Windows 11, 10 (64-bit), 8 (64-bit), Windows Server 2022, 2019, 2016, 2012, Server 2008 (64-bit), 7 (64-bit) and Vista (64-bit).

"C:\Program Files (x86)\SDI\TN3270 Plus\tn3270.exe" /Host 192.168.1.5:23

Script Switch Example:

When "/Script" switch is included, TN3270 Plus runs the specified script.

Windows 10 (32-bit), 8 (32-bit), 7 (32-bit), XP and Windows Server 2008(32-bit), 2003 and 2000.

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" /Session "**P/390 27x132"** /Script **logon.txt**

Windows 11, 10 (64-bit), 8 (64-bit), Windows Server 2022, 2019, 2016, 2012, Server 2008 (64-bit), 7 (64-bit) and Vista (64-bit).

"C:\Program Files (x86)\SDI\TN3270 Plus\tn3270.exe" /Session "**P/390 27x132"** /Script **logon.txt**

Where:

P/390 27x132 is the session name.
logon.txt is the script name.

Releases Earlier Than 2.4.0

TN3270 Plus releases earlier than 2.4.0 did not support command line switches. however, you could include a session name, script filename, web browser TN3270 command, web browser telnet command, domain name and port number or an IP address and port number on the command line. TN3270 Plus 2.4.0 still supports the older command line processing. For example:

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" "session_name"
"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" "script_filename"
"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" TN3270://hostname:port
"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" TELNET://hostname:port
"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" hostname**:**port

If a command line parameter is found, TN3270 Plus checks the parameter for a match in the following order:

1. Saved session name
2. Script filename
3. Web browser TN3270 command (TN3270://hostname:port/)
4. Web browser telnet command (TELNET://hostname:port)
5. If it's none of the above, then it's assumed to be a hostname:port.

Session Name Example:

When the session name is included, TN3270 Plus connects to that session when it starts.

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" "**P/390 27x132"**

Where:

P/390 27x132 is the session name.

Tips:

- The session name is case sensitive.
- If the session name contains any spaces, it must be enclosed in double quotes as in the example above.

Script Name Example:

When the script name is included, TN3270 Plus runs the specified script.

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" "**script.txt**"

Where:

script.txt is the script filename.

Tips:

- If the script filename contains any spaces, it must be enclosed in double quotes.

Web Browser TN3270 Command Example

This format is used when TN3270 Plus is specified as the default TN3270 application in your web browser options. When your web browser is pointed to a TN3270 URL, it starts TN3270 Plus with the URL on the command line. TN3270 Plus starts a TN3270 session for that URL.

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" **TN3270://TN3270site:23/**

Web Browser Telnet Command Example

This format is used when TN3270 Plus is specified as the default telnet application in your web browser options. When your web browser is pointed to a telnet URL, it starts TN3270 Plus with the URL on the command line. TN3270 Plus starts a VT100 session for that URL.

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" **TELNET://TelnetSite:23/**

Domain Name and Port Example:

When the domain name or IP address and port number are included, TN3270 Plus connects to the specified domain name or IP address on the specified port number.

"C:\Program Files\SDI\TN3270 Plus\tn3270.exe" domain.name:23

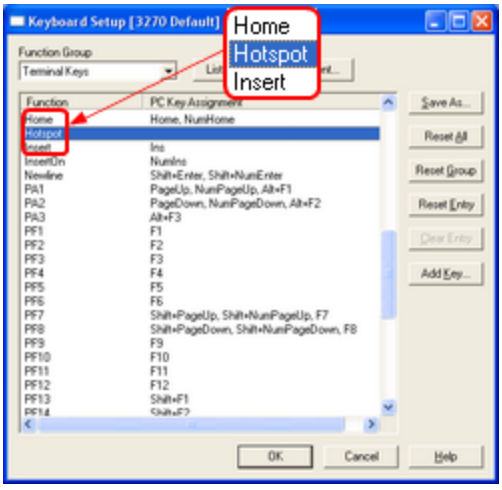
3.13 Using Hotspots

Hotspot support allows words on the terminal screen to act as AID generating keys, active URL links, active e-mail links, active UNC links, active file links or a launch pad for a script. This powerful support allows users to click on words on the terminal screen to cause an action. For example:

- You user could click on the word "F1" to rather than pressing the F1 key.
- You could click on "http://companyname.com" to activate the web browser and open the company web site.
- You could click on "support@sdisw.com" to start your e-mail client and pass it the e-mail address.
- You could click on file://c:/test.txt to open a file.
- You could click on \\servername\sharename\file.ext to open a shared file.
- If there is a script that corresponds to a word on the screen, then the script is run when the word is clicked. For example, clicking on "Help" would cause the "help.txt" script to run.

To activate Hotspot support, you need to assign the Hotspot terminal function to a mouse button as follows:

1. Open the **Setup** menu and select **Keyboard...**
2. In the Keyboard pane of the **Session Setup** dialog box, click the **Configure...** button.
3. In the **Keyboard Setup** dialog box, select "Terminal Keys" in the Function Group drop-down list box.
4. Select **Hotspot** in the Function list box.



5. Click the **Add Key...** button.
6. Click the desired mouse button over the text in the **Type Key** dialog box.
7. Click the **OK** button in the **Type Key** dialog box.

8. Click the **OK** button in the **Keyboard Setup** dialog box.
9. Click the **OK** or **Apply** button in the **Session Setup** dialog box.

Once the mouse button is assigned, clicking that button causes the word under the mouse pointer to be examined and appropriate actions is taken. The following type of hotspots are supported.

Hotspot type	Action
AID key	If the hotspot is an AID generating key, then the corresponding AID key is sent to the host.
Web site	If the hotspot begins with 'http://', 'https://' or "www." then the default web browser is launched and passed that URL.
E-mail address.	If the hotspot starts with "mailto:" or looks like an e-mail address then the default e-mail client is launched and passed the e-mail address.
UNC link	If the hotspot starts with \\servername\ then the file name is sent to the windows shell to be opened.
File link	If the hotspot starts with file:/// then the file name is sent to the windows shell to be opened.
Script	If the word is not one of the above, then a check is made to see if a script with that word name exists (wordname.txt). If found, the script is executed.

If a Hotspot is not recognized then the cursor is moved to the mouse pointer location and the ENTER key is sent to the host (simulating a menu selection).

An AID key identifier is delimited by a space, null, or attribute byte at the beginning and a space, null, attribute byte or non alpha-numeric character at the end. An e-mail address, URL, UNC or file link or or word is delimited by a space, null or attribute byte at the beginning and the end.

AID Generating Keys:

3270: PF1 - PF24, PA1 - PA3, CLEAR or ENTER
5250: F1 - F24, PA1 -PA3, CLEAR, ENTER or HELP

3.14 Using SDI LPD

SDI LPD is a 32-bit print daemon that accepts print jobs from a remote computer and prints them on a Windows printer. Complete details can be found in the SDI LPD User Guide.

3.15 Using the Start PC Command (STRCCMD)

TN3270 Plus supports the Start PC Command (STRPCCMD) for 5250 terminal sessions. The STRPCCMD command is part of the PC Organizer facility available on iSeries systems. TN3270 Plus does not support any of the other PC Organizer facility commands.

The STRPCCMD launches a Windows application on the PC that is running the TN3270 Plus 5250 terminal session. The STRPCCMD command can be invoked from the 5250 terminal session command line.

For example:

```
STRPCCMD PCCMD('start http://www.google.com') PAUSE(*NO)
```

3.16 Working with Multiple Sessions

There are two ways to use multiple sessions in TN3270 Plus

- Multiple sessions in a single window
- Multiple sessions in multiple windows

Creating Your Sessions

For either method of running multiple sessions, the first step is to create your sessions. Assume we need the following sessions:

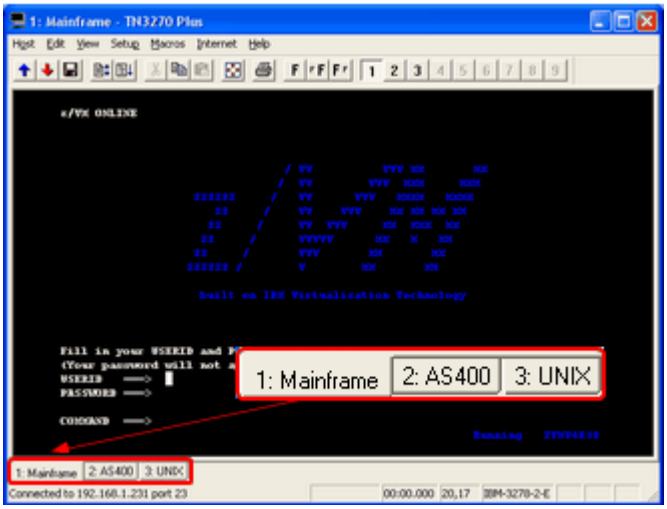
Session Name	Host Name	Telnet port	Terminal Type
Mainframe	tnserver.di3270.com	23	3270
AS400	as400.com	23	5250
UNIX	140.147.254.3	23	VT220

To create the sessions:

1. Open the **Connect to Host** dialog box (Host, Connect…).
2. Type in the settings for the first session (for example, Mainframe)
3. When the first session settings are complete, type in the session name for the next session (for example, AS400) and press the tab key. You will be prompted to save changes to the Mainframe session. Respond yes. Complete the settings for the AS400 session and press the tab key. You will be prompted to save changes to the AS400 session. Respond yes. Continue in the same manner until all your sessions are created.

Using Multiple Sessions in a Single Window

TN3270 Plus allows you to connect up to 99 different display and/or printer sessions concurrently in a single window. The display and/or printer sessions may be of different terminal types and may be connected to one or more host computers. Each session is given a session number between 1 and 99.

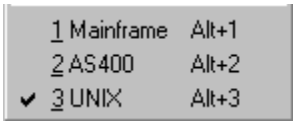


To connect the Mainframe, AS400 and UNIX sessions.

1. Open the **Connect to Host** dialog box (Host, Connect…).
2. Select **Mainframe** in the Session name list box.

3. Click the **Connect…** button.
4. To connect the AS400 session as your second session, open the **Connect to Host** dialog box (Host, Connect…).
5. Select **AS400** in the Session name list box.
6. Click the **Connect…** button.
7. Repeat the same steps to start the UNIX session.

Although multiple sessions may be active, only the current session is displayed in the TN3270 Plus window. To switch to another session, do one of the following:

	Select a new session tab. Session tabs appear after more than one session is connected. You can turn off session tabs (Setup, Preferences..., Don't show session tabs (requires restart).
	Select a new session button from the toolbar. Active session buttons are bold. Unused sessions are grayed out.
	Select a new session from the session list in the Host menu. The current session is marked with a check mark.
HostSession + n	Press the HostSession + n key (default Alt +n), where n is the session number.
ViewNextSesson	Press the ViewNextSession key (default Ctrl + Tab) to cycle through the sessions.
ViewPreviousSession	Press the ViewPreviousSession key (default Shift + Ctrl + Tab) to cycle backwards through the sessions.

Tips:

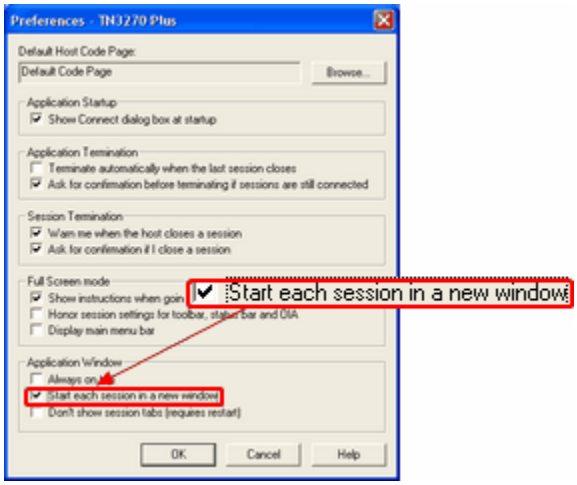
- If your want one or more sessions connected each time TN3270 Plus is started, specify the "Connect at Startup" option for the session(s). (Host, Connect…, Advanced button, Setup Items = Host, Connect at Startup),
- If you want a session started using the same session number every time you connect, you can specify a Preferred Session Number in the Session Setup dialog box. (Host, Connect, Advanced button, Setup Items = Host, Preferred Session Number) If the preferred session number is already in use, your session starts using the first available session number.
- By default TN3270 Plus allows 99 sessions in a single window, you can limit the number of sessions by adding the following DWORD registry entry:

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\MaximumSessions

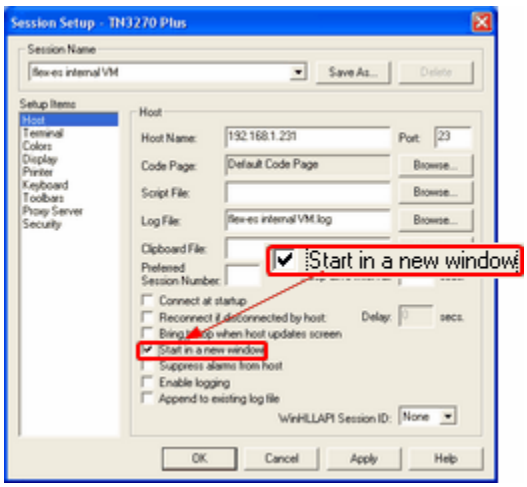
and setting it to the maximum number of sessions you wish to allow. See Limiting the Number of TN3270 Plus Sessions for more details.

Using Multiple Sessions in Multiple Windows

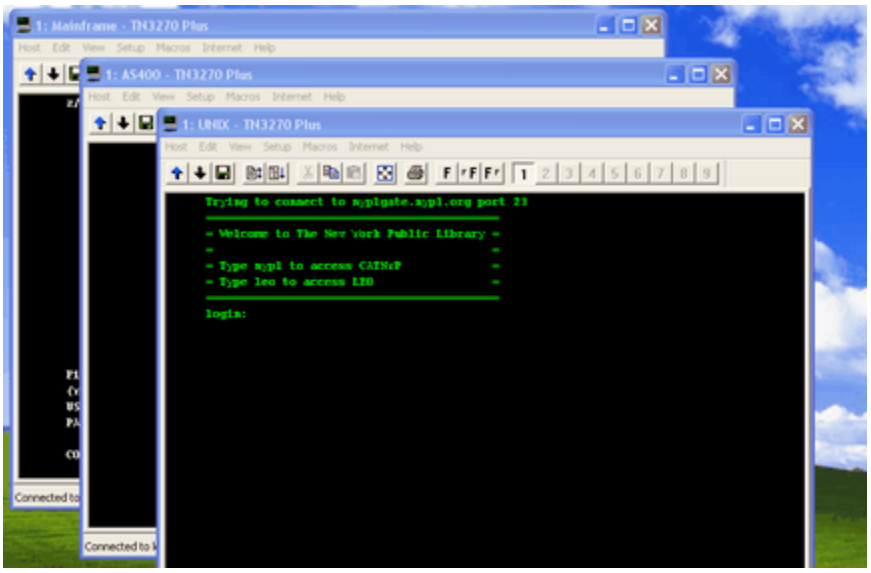
If you want to have multiple sessions in multiple Windows, specify the "Start each session in a new window" preference (Setup, Preferences…, Start each session in a new window).



If you only want some sessions to run in a separate window, you can use the "Start in a new window" session option (Setup, Sessions…, Setup Items = Host, Start in a new window).



For example, our sample sessions would like something like the following screen shot when run in multiple Windows instead of a single window.



If you want to create shortcut on your desktop to start TN3270 Plus and connect a particular session, do the following.

1. Create a TN3270 Plus shortcut on your desktop by right-clicking and dragging the TN3270 Plus icon to your desktop and dropping it.
2. Select "Create Shortcut(s) Here" from the pop-up menu.
3. Right-click the newly created desktop icon and select rename. Rename the icon as desired. (For example, TN3270 Plus Mainframe Session.)
4. Right-click the icon again and select Properties. In the Target edit box, add the session name after the path. For example, to add the Mainframe session:

   "C:\Program Files\SDI\TN3270 Plus\tn3270.exe" /Session "Mainframe"

   Double-clicking on this icon starts TN3270 Plus and connects the Mainframe session. For more information on specifying options for a shortcut icon, see Specifying Parameters on the Command Line.

Tips:

- By default TN3270 Plus does not limit the number of times it can be started. You can limit the number of times TN3270 Plus can be started (instances) by adding the following DWORD registry entry:

  HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\MaxInstances

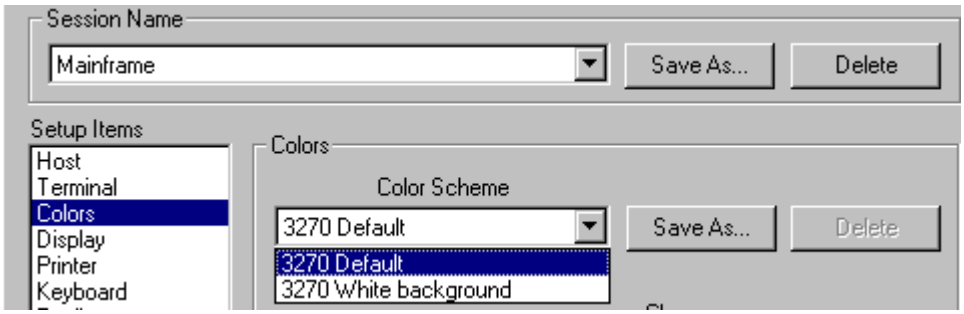
  and setting it to the maximum number of instances you wish to allow. When the maximum number of instances is reached an attempt to start another instance results in the most recently started instance being brought to the foreground. See Limiting the Number of TN3270 Plus Instances for more details.

3.17 Changing Session Settings

3.17.1 Changing Terminal Emulation Session Colors

1. Open the **Setup** menu and select the **Colors...** command.
2. Use the Colors Pane to define your color scheme.
3. If you wish to create a new color scheme with your changes, click the **Save As** button and type the new color scheme name in the **Save As** dialog box.

4. Click **OK** or **Apply** to save the changes to the current color scheme and update the colors in the current terminal emulation session.



3.17.2 Changing Screen Fonts

1. Open the **Setup** menu and select the **Display...** command.
2. Click on the **Change...** button to display the Font dialog box.
3. Select the font, font style, font size and font script.
4. The size of the window adjusts to accommodate the font you select.

Tips:

- TN3270 Plus can use any of the fixed-pitch fonts installed on your Windows system. In a fixed-pitch font, all the characters in the font are the same width.
- If a scalable font is selected, the "Scale font to fit terminal window" option is set on. This option resizes the font when the window is resized.
- The font remains in effect for the session even if you exit and restart TN3270 Plus.

3.17.3 Changing PC Keyboard and Mouse Mapping

PC Keyboard mapping allows you to customize your PC keyboard and mouse for terminal emulation sessions and for menu short cuts.

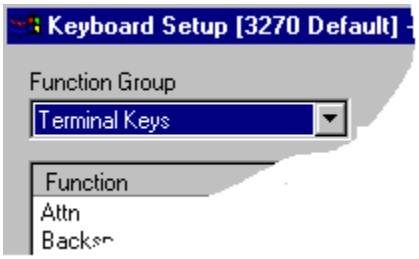
1. Open the **Setup** menu and select the **Sessions...** command to display the Session Setup dialog box.
2. In the **Session Setup** dialog box, select the **Session Name** of the session you want to change.
3. In the **Setup Items** list box, select **Keyboard** to display the Keyboard Pane.
4. In the **Keyboard Map Name** drop-down list box, select the desired keyboard map.
5. To modify an existing keyboard map or create a new keyboard map, click the **Configure...** button to display the Keyboard Setup dialog box. Once you have created a customized keyboard map, you can select it from the **Keyboard Map Name** drop-down list box for use with any of your sessions.

3.17.4 Changing VT100/VT220 Control Sequences

The default VT100/VT220 control sequences work for most host systems. However, if the requirements for your system are different, you can modify the default control sequences.

1. Open the **Setup** menu and select the **Sessions...** command to display the Session Setup dialog box.
2. In the **Session Setup** dialog box, select the **Session Name** of the session you want to change.
3. In the **Setup Items** list box, select **Keyboard** to display the Keyboard Pane.
4. In the **Keyboard Map Name** drop-down list box, select the desired keyboard map.

5. To modify an existing keyboard map or create a new keyboard map, click the **Configure...** button to display the Keyboard Setup dialog box. Once you have created a customized keyboard map, you can select it from the **Keyboard Map Name** drop-down list box for use with any of your sessions.
6. In the **Keyboard Setup** dialog box, select Terminal Keys in the **Function Group** drop-down list box.



7. Select the function you want to change in Function column of the list box.
8. The "Ctrl Seq:" label, **Change** button and Control Sequence edit box appear in the lower left-hand corner of the dialog box. To change the control sequence, position the cursor in the edit box and modify the control sequence. Data can be entered in character, hexadecimal or octal format. To enter character data in the edit box, just type the character(s). To enter hexadecimal data in the edit box, precede each byte (two hexadecimal numbers) with "0x". To enter octal data in the edit box, type precede each byte (three octal numbers) with a back slash (\). For example, to enter the hexadecimal control sequence "FDFF" type: "0xfd0xff" or "\375\377". Click the **Change** button to complete the update.

3.17.5 Changing the 3270 Attention and System Request Keys

In 3270 terminal emulation, the 3270 attention and system request functions send telnet command strings to the host computer. By default, the telnet command strings are defined as follows:

Key	Telnet Command String	
	TN3270	TN3270E
Attention	FFF3	FFF4
System Request	FFF4	FFF5

In rare instances, the default setting may not work with your host computer. In this case, you may change the command string sent by either of these keys. You can do this for all sessions by adding a registry entry or for a specific session by adding a line to the session profile. TN3270 Plus searches for the telnet command strings in the following order:

1. Session profile
2. Registry
3. Default setting

Adding a Registry Entry

Use this method to change the telnet command string for all sessions that do not have a value set in the session profile. To specify a new telnet command string, use the following steps:

1. Back up the registry. You should always back up the registry before making modifications.
2. Run regedit.exe
3. Navigate through the following keys and folders: HKEY_CURRENT_USER\Software\SDI\TN3270 Plus
4. Right click on the TN3270 Plus folder and select New from the popup menu and Key from the second level popup menu.

5. A key name of **New Value #1** is created. Replace **New Value #1** with **3270** and press enter.
6. Right click on the 3270 folder and select **New** from the popup menu. Select **String Value** from the second level popup menu.
7. A key name of **New Value #1** is created. Replace **New Value #1** with **AttnKeyString** or **SysreqKeyString** and press enter.
8. Right click on the newly created value and select **Modify** from the popup menu.
9. In the **Edit String** dialog box, enter the telnet command string into the **Value data** edit box and click the OK button.

Adding a Session Profile Entry

Use this method to change the telnet command string for a specific session. The session profile file name is the session name with the .tsp extension. For example, if your session name is "IBM Mainframe" the corresponding session profile name is "IBM Mainframe.tsp." To add the telnet command string entry:

1. Edit the session profile with notepad.exe or another word processor.
2. Add the following line(s) to the end of the session profile:
   AttnKeyString=*xxxx*
   SysreqKeyString=*xxxx*

   Where:
   xxxx is the telnet command string. The string must be a hexadecimal value.
3. Save the updated session profile and restart TN3270 Plus.

3.18 Transferring Files

3.18.1 Transferring Files from the Host Computer to Your PC

Use this procedure for zSeries (mainframe) systems that support IND$FILE type transfers. This type of file transfer operation is supported for 3270 sessions only. If you are interested in FTP file transfers, see the SDI FTP User Guide.

1. Open the **Host** menu and select **File Transfer...**
2. Fill in the File Transfer dialog box.
3. Click the  button in the **File Transfer** dialog box to start the transfer.
4. To cancel an active file transfer, select **Cancel File Transfer** from the **Host** menu.

Tips:

- You may switch to, and work in, another session while the file transfer is active.
- It is also possible to transfer a file using a script. If you often transfer the same file, a script is a more automated way to perform the operation. See the FileTransfer script command for more information on file transfer operations using a script.

3.18.2 Transferring Files from Your PC to the Host Computer

Use this procedure for zSeries (mainframe) systems that support IND$FILE type transfers. This type of file transfer operation is supported for 3270 sessions only. If you are interested in FTP file transfers, see the SDI FTP User Guide.

1. Open the **Host** menu and select **File Transfer...**
2. Fill in the File Transfer dialog box.

3. Click the  button in the **File Transfer** dialog box to start the transfer.
4. To cancel an active file transfer, select **Cancel File Transfer** from the **Host** menu.

Tips:

- You may switch to, and work in, another session while the file transfer is active.
- It is also possible to transfer a file using a script. If you often transfer the same file, a script is a more automated way to perform the operation. See the FileTransfer script command for more information on file transfer operations using a script.

3.19 Using Macros and Scripts

3.19.1 Using the Macro Recorder

The TN3270 Plus macro recorder is designed to automate keystroke sequences. Use the macro recorder to record a sequence of keystrokes as you type them. Then replay the macro to have the same keystrokes typed for you by the macro recorder.

Here are some common uses for macros:

- Logging on terminal emulation users.
- Performing a series of commonly issued commands.

Normally you will start the macro recorder, perform a common sequence of keystrokes and stop the macro recorder. Once the macro is recorded it may be played anytime you require the same keystroke sequence.

The TN3270 Plus scripting language also provides the ability to automate common tasks. See the Macro Recorder and Scripting Language Comparison

for a discussion about which feature to use to automate your work.

Macro Syntax

A macro contains text and terminal key functions. Text appears in the macro with no punctuation. Keyboard function keys are represented by the function key name enclosed in a less than and greater than sign pair **<** ***keyname*** **>**. For example, the enter key appears in a macro as follows:

<Enter>

The complete list of ***keynames*** can be found in the **Keyboard Map Setup** dialog box. To display the **Keyboard Map Setup** dialog box, open the **Setup** menu and select **Sessions...,** from the **Setup Items** list box choose **Keyboard** and then click the **Configure...** button. Select Terminal Keys in the **Function Group** drop-down list box.

You may also consult the default keyboard maps in this help file:

Default 3270 Keyboard Map
Default 5250 Keyboard Map
Default VT100 Keyboard Map

Macro Recorder Tutorial

The following tutorial provides step-by-step instructions for recording the "sample" macro. The sample macro types the word "test" at the current cursor location and then "presses" the enter key. Connect a terminal emulation session, position the cursor to a data entry position and then follow the steps below.

Recording the Sample Macro

1. Open the **Macros** menu and select **Start Recording**.
2. Type "test" and press the enter key.
3. Open the **Macros** menu and select **Stop Recording**.
4. Enter the name "Sample" for the macro in the **Save As** dialog box.

5. Click the **OK** button.

Playing the Sample Macro

1. Return the session to the state it was in before you recorded the Sample macro.
2. Open the **Macros** menu and select **Replay**.
3. A second level menu listing of all recorded macros appears. Select the Sample macro.
4. The Sample macro runs. It should produce the same result as the keystrokes used to record the macro.

Editing the Sample Macro

1. Open the **Macros** menu and select **Edit**.
2. A second level menu listing of all recorded macros appears. Select the Sample macro.
3. The Macro Editor dialog box is displayed. The macro definition should contain the following:

   test<Enter>

4. Click OK without making any changes. See the Macro Editor dialog box topic for information on editing a macro.

Assigning a Keyboard Shortcut to the Sample Macro

1. Open the **Macros** menu and select **Assign Key**.
2. A second level menu listing of all recorded macros appears. Select the Sample macro.
3. The **Type Key** dialog box prompts you for the keyboard shortcut. Hold down the shift and alt keys and press the letter a. **Shift+Alt+A** appears in the edit box.
4. If **Shift+Alt+A** does not appear in the edit box, click the Retry button and try again.
5. Once the Sample macro is assigned to **Shift+Alt+A,** click the OK button.
6. Return the session to the state is was in before you recorded the Sample macro.
7. Press **Shift+Alt+A.** The Sample macro runs.

Deleting the Sample Macro

1. Open the **Macros** menu and select **Delete**.
2. A second level menu listing of all recorded macros appears. Select the Sample macro.
3. The macro is deleted.

Pausing Macro Execution

It is possible to pause macro execution if the macro needs to wait for a host response before continuing. See the Macro Editor dialog box topic for more information.

3.19.2 Creating a Logon Script (Release 3.6 and above)

One of the most common tasks for users is to logon to the host computer. This makes the logon process a good candidate for scripting. The following sample script illustrates logging on to a VM/ESA host. Use it as a starting point for automating your logon process.

```
'       z/VM Logon Script                          '1
waitfor("USERID:")                                 '2
type(test)                                         '3
key(tab)                                           '4
askfor("Enter your password and click OK",_
       password)                                   '5
key(enter)                                         '6
exit                                               '7
```

The following list describes each of the lines in this script.

1. Comment.
2. **waitfor("USERID:")** pauses the script until the characters "USERID:" appear on the emulation screen. This pause prevents the script from issuing more script commands before the host computer is ready to receive them. The parameter for the waitfor command is case sensitive. It must be typed exactly as it will appear on the emulation screen.
3. **type(Test)** types "Test" at the current cursor location.
4. **key(tab)** simulates pressing the tab key on the keyboard. This moves the cursor to the next tab stop.
5. **askfor("Enter your password and click OK",password)** displays a dialog box requesting the password. The password parameter at the end of the askfor command indicates that the characters typed by the user will be displayed as asterisks, so others cannot read the password. When OK is clicked, the characters entered in the dialog are typed at the current cursor location.
6. **key(enter)** simulates pressing of the enter key on the keyboard. This sends the data entered on the screen to the host computer for processing.
7. **exit** exits the script.

3.19.3 Creating a Logon Script (Release 3.5 and below)

One of the most common tasks for users is to logon to the host computer. This makes the logon process a good candidate for scripting. The following sample script illustrates logging on to a VM/ESA host. Use it as a starting point for automating your logon process.

```
*       z/VM Logon Script                    1
waitfor USERID:                              2
type    Test                                 3
key     tab                                  4
askfor  "Enter your password and click       5
        OK",password
key     enter                                6
exit                                         7
```

The following list describes each of the lines in this script.

1. Comment.
2. **waitfor USERID:** pauses the script until the characters "USERID:" appear on the emulation screen. This pause prevents the script from issuing more script commands before the host computer is ready to receive them. The parameter for the waitfor command is case sensitive. It must be typed exactly as it will appear on the emulation screen.
3. **type Test** types "Test" at the current cursor location.
4. **key tab** simulates pressing the tab key on the keyboard. This moves the cursor to the next tab stop.
5. **askfor "Enter your password and click OK",password** displays a dialog box requesting the password. The password parameter at the end of the askfor command indicates that the characters typed by the user will be displayed as asterisks, so others cannot read the password. When OK is clicked, the characters entered in the dialog are typed at the current cursor location.
6. **key enter** simulates pressing of the enter key on the keyboard. This sends the data entered on the screen to the host computer for processing.
7. **exit** exits the script.

3.19.4 Comparing the Macro Recorder and the Scripting Language

TN3270 Plus provides two ways to automate common tasks for users, the macro recorder and the scripting language.

The macro recorder records terminal keystrokes and allows you to replay them. It is quick and easy to use, but its function is limited in scope.

The scripting language can automate keystrokes and it can also contain logic statements, transfer files, prompt the user for input and start other applications. The scripting language requires that you create a text file containing script commands with a word processor and then run the script file. The scripting language is more difficult to use than the macro recorder.

If you just want to automate keystrokes, the macro recorder is the best choice. If you want to automate other tasks you will need to write a script.

The following table summarizes the facilities in the Macro Recorder and the Scripting Language.

Operation	Macro Recorder	Scripting Language
Record keystrokes	Yes	No
Type text	Yes	Yes
"Press" terminal key	Yes	Yes
Wait for a number of seconds	Yes	Yes
Wait for text from host	No	Yes
Conditional processing	No	Yes
Transfer files	No	Yes
Prompt user	No	Yes
Start other applications	No	Yes
Connect sessions	No	Yes
Switch sessions	No	Yes

4 Administrators Guide

4.1 Compatibility with Earlier Releases

SDI makes a substantial effort to insure conversion from one release of TN3270 Plus to another is a seamless as possible. Users should not notice any change unless they wish to take advantage of a new feature. For example, release 4.0 will honor all the configuration settings of earlier releases of TN3270 Plus. Simply uninstall the earlier release and install release 4.0. Users will see the same familiar interface. Modifications to keyboard maps, colors definitions and all other configuration settings remain intact. Any scripts or macros will continue to run without modification.

We are always working to make TN3270 Plus faster, more efficient and easier to use. We urge you to move to newer releases as they become available so that you can take advantage of new enhancements and features. You can see list of the updates included in TN3270 Plus release 4.0 on our Version History web page.

4.2 Configuration Files

TN3270 Plus keeps configuration information in its configuration files. The following table summarizes the contents of the TN3270 Plus configuration files.

File(s)	Description
*.tcs	Color scheme

*.tkm	Keyboard map
*.ttb	Toolbar
*.tsp	Session profile
*.txt	Script files
*.mac	Macros
*.pem	TLS certificate

The location of the configuration files can be found in the TN3270 Plus preferences (Setup, Preferences, Configuration Files Folder). If there is no entry for the Configuration Files Folder in the Preferences, see the Configuration Files Search Sequence below for the search sequence TN3270 Plus uses to locate its configuration files.

Configuration Files Search Sequence

It is important to understand the sequence TN3270 Plus uses to find the path to its configuration files. TN3270 Plus looks for the path to its configuration files using the following sequence. TN3270 Plus uses the first path it finds.

Windows 11, 10 (64-bit), 8 (64-bit), 7 (64-bit) Windows Server 2022, 2019, 2016, 2012, 2008 (64-bit), 2003(64-bit).

1. Configuration Files Folder (Setup, Preferences, Configuration Files Folder).
2. HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration\DataPath\
3. The working directory if one is specified and it contains one or more session profiles (.tsp files).
4. The install directory if it contains one or more session profiles (.tsp files).
5. The TN3270 Plus application directory (c:\users*user_id*\AppData\Roaming\SDI\TN3270 Plus\). This is the **default location**.

Windows 10 (32-bit), 8 (32-bit), 7 (32-bit), Windows Server 2008 (32-bit), and 2003 (32-bit)

1. Configuration Files Folder (Setup, Preferences, Configuration Files Folder).
2. HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\DataPath\
3. The working directory if one is specified and it contains one or more session profiles (.tsp files).
4. The install directory if it contains one or more session profiles (.tsp files).
5. The TN3270 Plus application directory (c:\users*user_id*\AppData\Roaming\SDI\TN3270 Plus\). This is the **default location**.

Windows XP and Windows Server 2000

1. Configuration Files Folder (Setup, Preferences, Configuration Files Folder).
2. HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\DataPath\
3. The working directory if one is specified.
4. The install directory. (C:\Program Files\SDI\TN3270 Plus\ if the default install directory was used.) This is the **default location**.

4.3 Disabling Menu Items

For security reasons, you may wish to restrict some users from issuing some menu commands. TN3270 Plus allows you to do so by adding registry entries to disable selected menu commands.

The following table lists registry entries you can use to disable menu commands. These entries can be placed in:

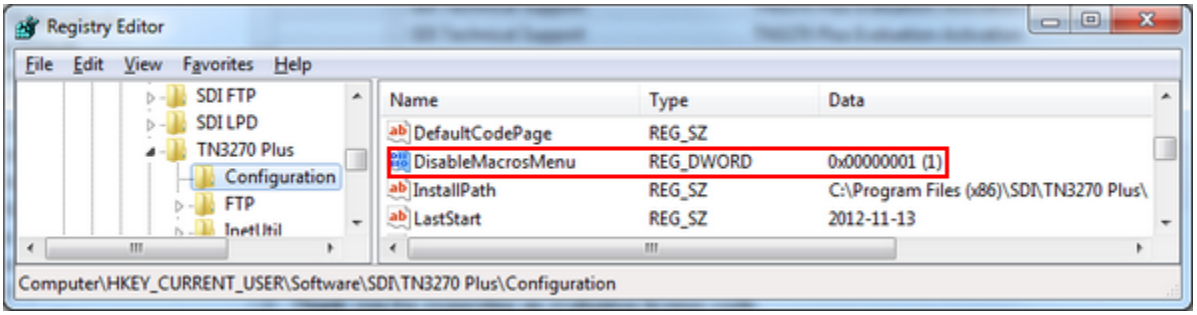
HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration
or

HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration (64-bit Windows)
or
HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration (32-bit Windows)

Placing the registry entry in the HKEY_CURRENT_USER hive restricts the entry to a specific user. Placing the registry entry in the HKEY_LOCAL_MACHINE hive restricts all users on the PC. All the entries in the table have DWORD (32-bit) values. Set the value to 1 to disable the menu item or 0 to enable it. The default is enabled for all menu items.

Registry entry	Menu Item(s)
DisableLanguageSelection	Language - Disables the Language menu.
DisableMacrosMenu	Macros - Disables the ability to record, edit or play macros.
DisableMacroRecording	Disables the ability to record or edit macros, but allows macros to be played.
DisableMenuBar	All - Removes the menu bar. No menu commands are available.
DisablePopupMenu	Context (Pop-up) - Disables the context menu.
DisableSessionChanges	Setup - Disables the Setup menu. Disables the Host, Enable Logging and Host, Disable Logging menu items. Also disables all controls in the **Connect to Host** dialog box. This prevents the user from making any changes to session options including connection information, terminal settings, keyboard mapping, color schemes, display settings, printer settings and security options.

The following screen image shows the DisableMacrosMenu registry entry:



4.4 Disabling Security Settings

For security reasons, you may wish to prevent users from changing the TN3270 Plus security settings for TLS or SSH encryption. For example, you may want to prevent users from changing the TN3270 Plus encryption protocol from TLS version 1.2 to TLS version 1.1.

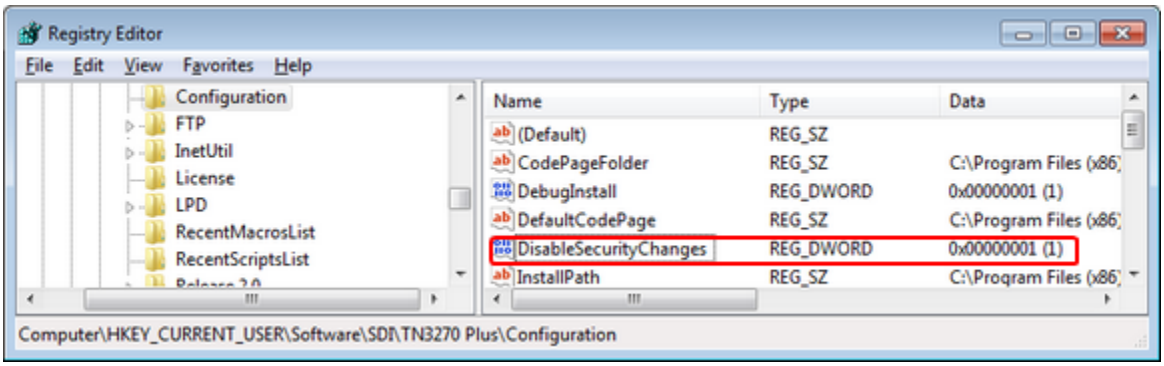
Add the DisableSecurityChanges registry entry to the registry to disable the TN3270 Plus security settings. This entry can be placed in any of the following:

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration
or
HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration (64-bit Windows)
or
HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration (32-bit Windows)

Placing the registry entry in the HKEY_CURRENT_USER hive restricts the entry to a specific user. Placing the registry entry in the HKEY_LOCAL_MACHINE hive restricts all users on the PC. The registry entry has a DWORD (32-bit) value. Set the value of this entry to 1 to disable security settings, 0 to enable them. The default is that security settings are enabled.

Disabling the security settings affects the controls in the Security pane of Session Setup dialog box and the "Connection Type" radio buttons in the Connect to Host dialog box. In the Security pane of Session Setup dialog box the Encryption Protocol edit box is disabled (grayed out) and all the other controls and hidden. In the Connect to Host dialog box the Telnet, TLSv1 and SSH radio buttons are and disabled (grayed out).

The following screen image shows and example of the DisableSecurityChanges registry entry:



4.5 Global Registry Settings

The following registry entries are read from HKEY_LOCAL_MACHINE if they are not found in HKEY_CURRENT_USER. This allows you to set global values for all users on a PC by moving these entries from HKEY_CURRENT_USER to HKEY_LOCAL_MACHINE.

DWORD Values

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\DisableEvaluationReminder
(1 = true, 0 = false)

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\MaxInstances

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\MaximumSessions

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\Preferences
(See the Preferences Command for settings.)

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\ShowDefaultAppDialog
(1 = true, 0 = false)

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Release *v.m*\Settings\DisableUpdateCheck
(1 = true, 0 = false)

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\DisableLanguageSelection
(1 = true, 0 = false)

String Values

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\DataPath

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\LicenseCodePath

Windows 11, 10 (64-bit), 8 (64-bit), 7 (64-bit) Windows Server 2022, 2019, 2016, 2012, 2008 (64-bit) and 2003 (64-bit)

If you are running a 64-bit version of Windows, use:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\SDI\TN3270 Plus\...

instead of:

HKEY_LOCAL_MACHINE\SOFTWARE\SDI\TN3270 Plus\...

Suggested Folders for Shared Data

The following folders are recommended for sharing the TN3270 Plus configuration files and/or licence code file between all users on a PC.

Windows 11, 10, 8, 7, Windows Server 2022, 2019, 2016 2012, 2008 and 2003

c:\ProgramData\SDI\TN3270 Plus\

Windows XP

c:\Documents and Settings\All Users\Application Data\SDI\TN3270 Plus

Windows 2000

c:\Program Files\SDI\TN3270 Plus\

4.6 Limiting The Number of TN3270 Plus Sessions

TN3270 Plus allows you to connect up to 99 different display and/or printer sessions concurrently in a single window. The display and/or printer sessions may be of different terminal types and may be connected to one or more host computers. Each session is given a session number between 1 and 99. For more information about using multiple sessions, see Working With Multiple Sessions.

By default TN3270 Plus allows 99 sessions in a single window, you can limit the number of sessions by adding the following DWORD registry entry:

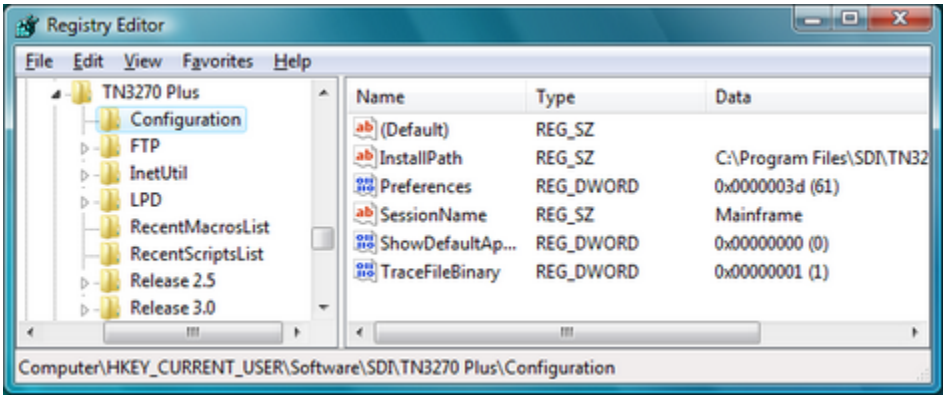
HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\MaximumSessions

and setting it to the maximum number of sessions you wish to allow. Numbers from 1 to 99 are valid.

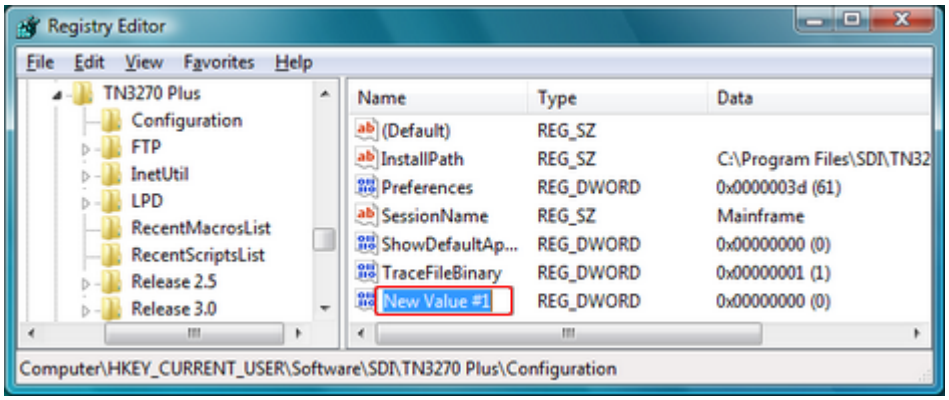
Step-By-Step

Use the following step-by-step instructions to make the registry change required to limit the TN3270 Plus sessions.

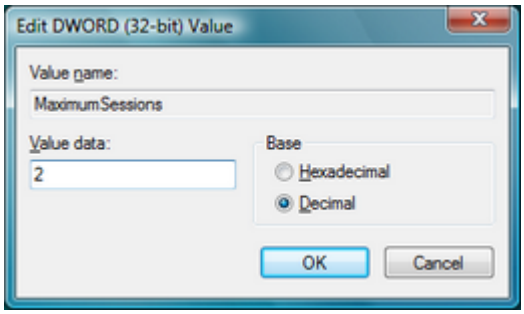
1. Run regedit.exe.
2. Navigate to HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\



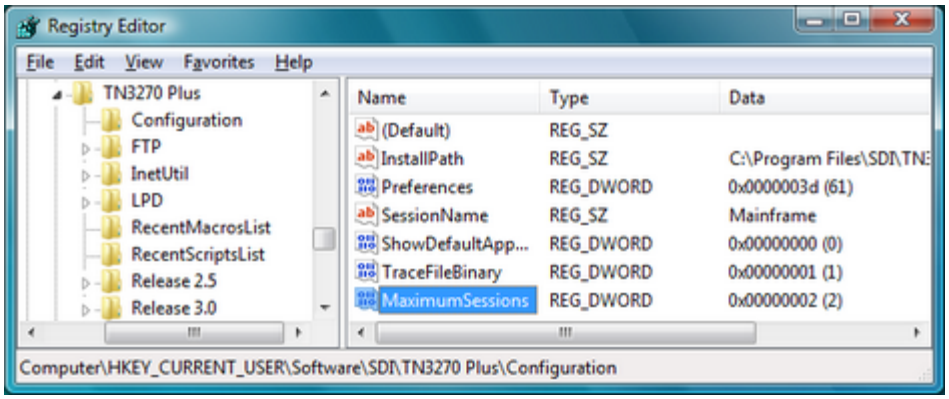
3. Right click on the Configuration key and expand the **New** context menu item and select **DWORD VALUE.** A "New Value #1" entry appear in the right pane. Rename this value to "MaximumSessions".



4. After you have renamed "New Value #1" to "MaximumSessions" right click on MaximumSessions and select **Modify** from the context menu to display the **Edit DWORD Value** dialog. Click on the **Decimal** radio button and then enter the maximum number of sessions you wish to allow in the **Value data** edit box. Numbers from 1 to 99 are valid.



5. Click the **OK** button to complete the update.



4.7 Limiting The Number of TN3270 Plus Instances

By default TN3270 Plus does not limit the number of times it can be started. You can limit the number of times TN3270 Plus can be started (instances) by adding the following DWORD registry entry:

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\MaxInstances

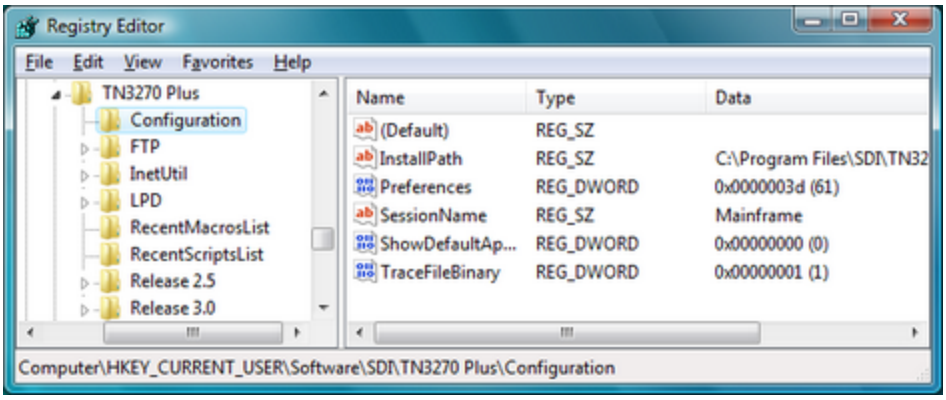
and setting it to the maximum number of instances you wish to allow. When the maximum number of instances is reached an attempt to start another instance results in the most recently started instance being brought to the foreground. If the value is set to zero, TN3270 Plus will not start. If the value is set to –1, unlimited instances are

allowed.

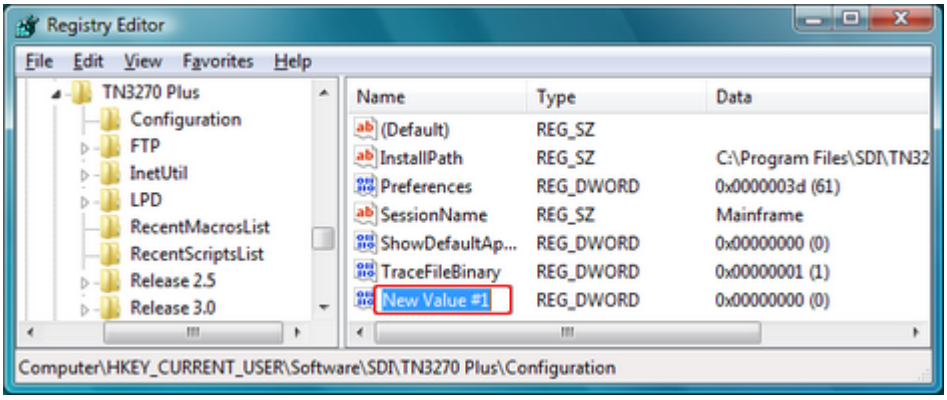
Step-By-Step

Use the following step-by-step instructions to make the registry change required to limit the TN3270 Plus instances.

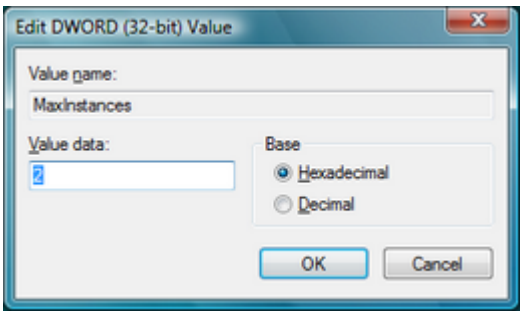
1. Run regedit.exe.
2. Navigate to HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\



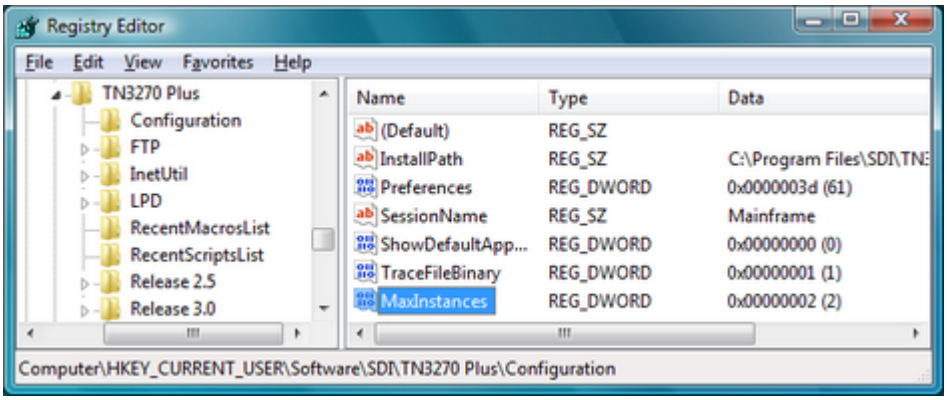
3. Right click on the Configuration key and expand the **New** context menu item and select **DWORD VALUE.** A "New Value #1" entry appear in the right pane. Rename this value to "MaxInstances".



4. After you have renamed "New Value #1" to "MaxInstances" right click on MaxInstances and select **Modify** from the context menu to display the **Edit DWORD Value** dialog. Click on the **Decimal** radio button and then enter the maximum number of instances you wish to allow in the **Value data** edit box.



5. Click the **OK** button to complete the update.



4.8 Lockdown Session Settings

TN3270 Plus lockdown prevents users from making changes to their sessions.

The lockdown option disables the following items:

1. The TN3270 Plus **Setup** menu.
2. All controls in the **Connect to Host** dialog box. This prevents the user from making any changes to session options including connection information, terminal settings, keyboard mapping, color schemes, display settings, printer settings and security options.
3. The "Enable Logging" and "Disable Logging" commands on the **Host** menu.

Important Note

Please use the new DisableSessionSettings registry key for locking down session settings instead the the Preferences key referenced in this topic. The technique below is no longer recommended and is only documented here for backward compatibility.

The lockdown option is enabled by adding 0x00000800 (2048 decimal) to the value in one of the following registry keys.

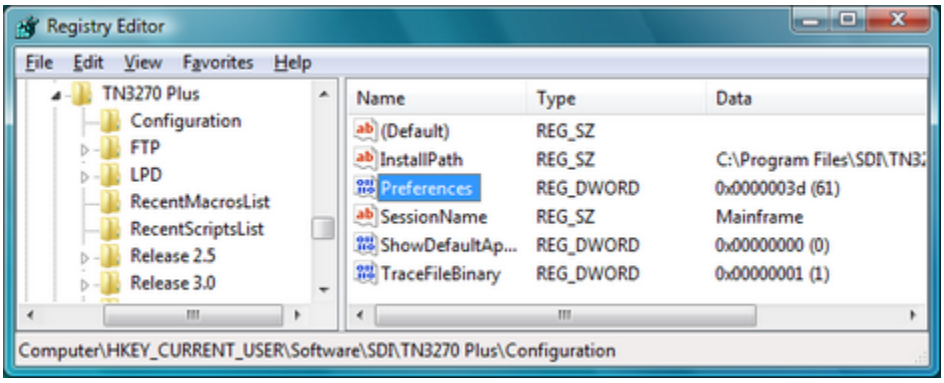
HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\Preferences
HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration\Preferences (64-bit Windows)
HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\Preferences (32-bit Windows)

The HKEY_CURRENT_USER entry only affects the current user. The HKEY_LOCAL_MACHINE entry affects all users on the PC. If entries are specified in both locations the HKEY_CURRENT_USER entry takes precedence.

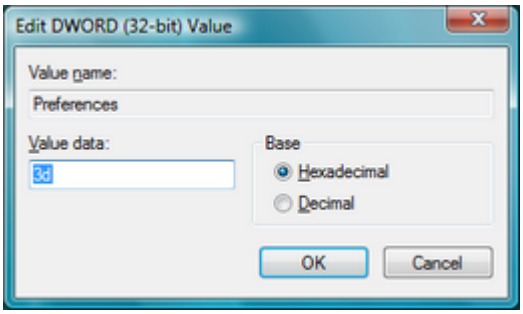
Step-By-Step

Use the following step-by-step instructions to make the registry change required to lockdown the TN3270 Plus settings.

1. Run regedit.exe.
2. Navigate to HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\Preferences



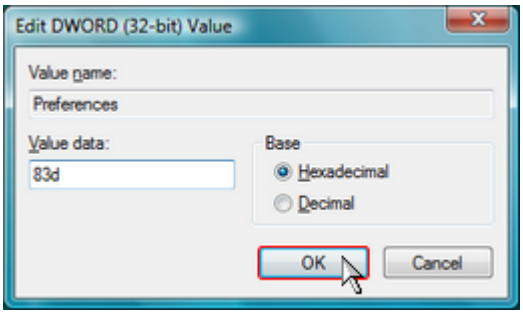
3. Right click on the Preferences key and select **Modify** to display the **Edit DWORD Value** dialog box.



4. If you are comfortable with hexadecimal addition, add 0x800 to the value in the "Value data" edit box. For example 0x3d + 0x800 = 0x83d.

   If you are not comfortable with hexadecimal addition, click on the "Decimal" radio button and convert the "Value data" to a decimal value. Then add 2048 to the decimal value. For example, 61 + 2048 = 2109.

5. Click the **OK** button to complete the update.



Disable Standard and Popup Menus

It is also possible to disable the TN3270 Plus standard menu and the popup menu. The standard menu is disabled by adding 0x00008000 to the existing value of one of the following registry keys. The popup menu is disabled by by adding 0x00010000 to the existing value of one of the following registry keys.

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\Preferences
HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration\Preferences (64-bit Windows)
HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\Preferences (32-bit Windows)

The HKEY_CURRENT_USER entry only affects the current user. The HKEY_LOCAL_MACHINE entry affects all users on the PC. See Preferences for more details.

See Also:

GlobalRegistrySettings
Preferences command
Preferences

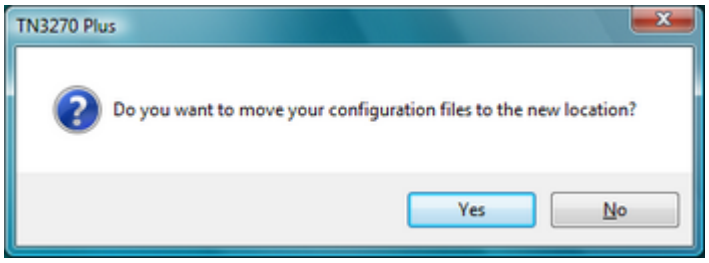
4.9 Moving the TN3270 Plus Configuration Files

TN3270 Plus keeps configuration information in its configuration files. Follow these steps to move the configuration files to a new location.

Step-By-Step

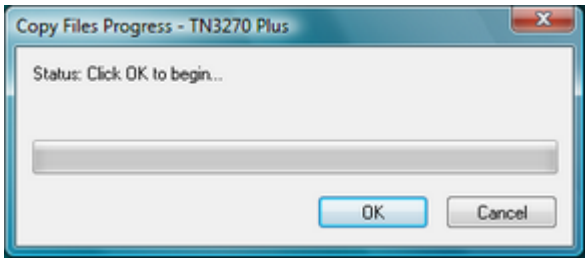
Use the following instructions to move (or change the location of) your TN3270 Plus configuration files.

1. Close all TN3270 Plus sessions.
2. Open the **Setup** menu and select **Preferences...** to display the **Preferences** dialog box.
3. Change the "Configuration Files Folder" edit box to the new folder location. You may type in a new location or use the **Browse...** button to select a new location.
4. Click the **OK** button.
5. If there are no configuration files in the new folder TN3270 Plus will ask you if you want to move your existing files to the new folder.



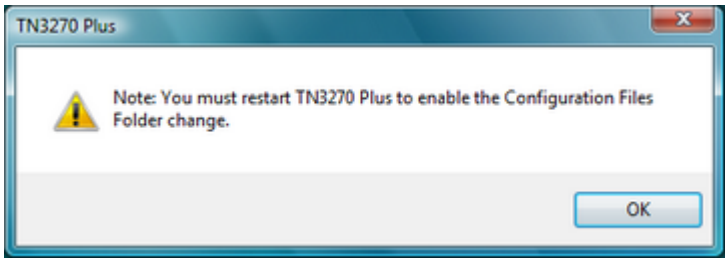
Click the **Yes** button to copy your configuration files to the new folder.

The **Copy Files Progress** dialog appears.



Click the **OK** button to to start the copy operation. Click the **OK** button again when the copy operation is complete to dismiss the dialog. (The configuration files in the old folder are not deleted.)

6. TN3270 Plus prompts you to restart.



Click the **OK** button to dismiss the dialog.

7. Exit TN3270 Plus.
8. Restart TN3270 Plus to access the new configuration files folder.

4.10 Moving the License Code File

When the TN3270 Plus license code is entered, it is stored in the tn3270.ini file. The default location for the tn3270.ini file is in the /SDI/TN3270 Plus/ sub-folder in the Windows common application data folder. Here is the default folder location for the different Windows operating systems:

11/10/8/7 2022/2019/2016/2012/2008	C:\ProgramData\SDI\TN3270 Plus\
2003/XP/2000	C:\Documents and Settings\All Users\Application Data\SDI\TN3270 Plus\

If you wish to store the tn3270.ini file in a different folder you can add one of the following string value registry entries and set its value to the full path to the folder you want to contain the tn3270.ini file:

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\LicenseCodePath\

HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\LicenseCodePath\

The HKEY_CURRENT_USER entry only affects the current user. The HKEY_LOCAL_MACHINE entry affects all users on the PC. If entries are specified in both locations the HKEY_CURRENT_USER entry takes precedence.

Windows 11, 10 (64 bit), 8 (64 bit), 7 (64-bit), Windows Server 2022, 2019, 2016, 2012, 2008 (64-bit)

If you are running a 64-bit version of Windows, use:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\SDI\TN3270 Plus\...

instead of:

HKEY_LOCAL_MACHINE\SOFTWARE\SDI\TN3270 Plus\...

4.11 Moving TN3270 Plus from one PC to Another

Use this procedure to move TN3270 Plus and its configuration settings from one PC (old PC) to another (new PC).

You must contact SDI for permission prior to moving a TN3270 Plus license from one workstation to another as indicated in paragraph 4 of the SDI Product End User License Agreement (EULA).

Step-By-Step

On the old PC:

1. Locate the TN3270 Plus configuration files folder. This folder can be found by opening the **Setup** menu and selecting **Preferences...** to display the **Preferences** dialog box. In the **Preferences** dialog box, you will see the configuration file folder in the "Configuration Files Folder" edit box.
2. Copy all the files in the TN3270 Plus Configuration Files Folder to a removable storage device or a network drive accessible by the new PC.

On the new PC:

3. Download the current version of TN3270 Plus using the following link: http://sdisw.com/download.aspx
4. Execute the downloaded distribution file to start the installation wizard and install TN3270 Plus.
5. Start TN3270 Plus (Start, All Programs, TN3270 Plus, TN3270 Plus) and enter your license code by clicking the "I have a license code" button in the **Welcome to TN3270 Plus** dialog box.
6. Locate the TN3270 Plus configuration files folder as you did in Step 1. If you are installing on a different version of Windows, the configuration files folder may not be in the same location as it was on the old PC.
7. Copy all the configuration files saved in step 2 to the TN3270 Plus Configuration Files Folder.
8. Restart TN3270 Plus.

4.12 Preferences

Users can update TN3270 Plus preferences by opening the Setup menu and selecting Preferences. See the Preferences Command for a description of each preference setting.

The TN3270 Plus user preferences are kept in the following doubleword registry key:

HKEY_CURRENT_USER\Software\SDI\ TN3270 Plus\Configuration\Preferences

By default, the preference settings are user specific. Administrators can apply preferences to all users and also have the ability to set some preferences that are not available to the user.

Administrators can set preferences for all users on the PC by copying the doubleword value from
HKEY_CURRENT_USER\Software\SDI\ TN3270 Plus\Configuration\Preferences
to
HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\ TN3270 Plus\Configuration\Preferences (64-bit Windows)
HKEY_LOCAL_MACHINE\Software\SDI\ TN3270 Plus\Configuration\Preferences (32-bit Windows)

and then deleting
HKEY_CURRENT_USER\Software\SDI\ TN3270 Plus\Configuration\Preferences.

The Preferences key is a doubleword and uses the following hexadecimal values: To add a preference, add its value to the current Preference value.

Application Startup	
Show Connect at startup	0x00000001
Application Termination	
Terminate automatically when the last session closes	0x00000002
Ask for confirmation before terminating if sessions are still connected.	0x00000010
Session Termination	
Warn me when the host closes a session	0x00000004
Ask for confirmation if I close a session	0x00000008
Full Screen Mode	
Show instruction when going to full screen mode	0x00000020
Honor session setting for toolbar, status bar and OIA	0x00000040

Display main menu bar	0x00000080
Application Window	
Always on top	0x00000100
Start each session in a new window	0x00000400
Don't show session tabs (requires restart)	0x00000200
Show session tabs above terminal window (require restart)	0x00004000

Additional Preferences

The following preferences are not included in the **Preferences** dialog box, but they can be changed by adding the following value(s) to the Preferences registry key referenced above:

Disable menu bar	0x00008000
Disable popup menu	0x00010000

It is suggested that you use the DisableMenuBar and DisablePopupMenu registry keys instead the the Preferences key referenced in this topic. The technique above is no longer recommended and is only documented here for backward compatibility.

4.13 Printer Session Timeout

You can set a printer session timeout value by adding the following parameter to the printer session's profile (*session_name*.tsp),

```
PrinterReceiveTimeout=nn
```

Where:

nn	is the timeout value in seconds. The default is 30 seconds if omitted. The timeout value is used to force a PRINT-EOJ if no PRINT-EOJ indicator is received from the host within the timeout period.

This entry must be manually added to the end of the session profile file. The session profile is a text file and you can edit it using notepad.exe. The session profile is located in the Configuration Files Folder (Setup, Preferences..., Configuration Files Folder).f

4.14 TLS Cipher Suites

TN3270 Plus TLS (Transport Layer Security) supports the following cipher suites. A cipher suite is a named combination of key exchange, authentication, encryption and message authentication code (MAC) algorithms.

The cipher details for an active TLS connection can be viewed in the Server Certificate Details dialog box.

Cipher Suite	Key Exchange	Key Authentication	Encryption	MAC
ECDHE-RSA-AES256-GCM-SHA384	ECDHE	RSA	AES256-GCM	SHA256
ECDHE-ECDSA-AES256-GCM-SHA384	ECDHE	ECDSA	AES256-GCM	SHA384
ECDHE-RSA-AES256-SHA384	ECDHE	RSA	AES256	SHA384
ECDHE-ECDSA-AES256-SHA384	ECDHE	ECDSA	AES256	SHA384
ECDHE-RSA-AES256-SHA	ECDHE	RSA	AES256	SHA
ECDHE-RSA-AES256-SHA	ECDHE	ECDSA	AES256	SHA

SRP-DSS-AES-CBC-SHA	SRP	DSS	AES-CBC	SHA
SRP-RSA-AES-256-CBC-SHA	SRP	RSA	AES-256-CBC	SHA
SRP-AES-256-CBC-SHA	SRP		AES-256-CBC	SHA
DHE-DSS-AES256-GCM-SHA384	DHE	DSS	AES256-GCM	SHA384
DHE-RSA-AES256-GCM-SHA384	DHE	RSA	-AES256-GCM	SHA384
DHE-RSA-AES256-SHA256	DHE	RSA	AES256	SHA256
DHE-DSS-AES256-SHA256	DHE	DSS	AES256	SHA256
DHE-RSA-AES256-SHA	DHE	RSA	AES256	SHA
DHE-DSS-AES256-SHA	DHE	DSS	AES256	SNA
DHE-RSA-CAMELLIA256-SHA	DHE	RSA	CAMELLIA256	SHA
DHE-DSS-CAMELLIA256-SHA	DHE	DSS	CAMELLIA256	SHA
ECDH-RSA-AES256-GCM-SHA384	ECDH	RSA	AES256-GCM	SHA384
ECDH-ECDSA-AES256-GCM-SHA384	ECDH	ECDSA	AES256-GCM	SHA384
ECDH-RSA-AES256-SHA	ECDH	RSA	AES256	SHA
ECDH-ECDSA-AES256-GCM-SHA	ECDH	ECDSA	AES256-GCM	SHA
AES256-GCM-SHA384	RSA	RSA	AES256-GCM	SHA384
AES256-SHA256	RSA	RSA	AES256	SHA256
AES256-SHA	RSA	RSA	AES256	SHA
CAMELLIA256-SHA	RSA	RSA	CAMELLIA256	SHA
PSK-AES256-CBC-SHA	PSK	PSK	AES256-CBC	SHA
ECDHE-RSA-AES128-GCM-SHA256	ECDHE	RSA	AES128-GCM	SHA256
ECDHE-ECDSA-AES128-GCM-SHA256	ECDHE	ECDSA	AES128-GCM	SHA256
ECDHE-RSA-AES128-SHA256	ECDHE	RSA	AES128	SHA256
ECDHE-ECDSA-AES128-SHA256	ECDHE	ECDSA	AES128	SHA256
ECDHE-RSA-AES128-SHA	ECDHE	RSA	AES128	SHA
ECDHE-ECDSA-AES128-SHA	ECDHE	ECDSA	AES128	SHA
SRP-DSS-AES-128-CBC-SHA	SRP	DSS	AES-128	SHA
SRP-RSA-AES-128-CBC-SHA	SRP	RSA	AES-128	SHA
SRP-AES-128-CBC-SHA	SRP	SRP	AES-128-CBC	SHA
DHE-DSS-AES128-GCM-SHA256	DHE	DSS	AES128-GCM	SHA256
DHE-RSA-AES128-GCM-SHA256	DHE	RSA	AES128-GCM	SHA256
DHE-RSA-AES128-SHA256	DHE	RSA	AES128	SHA256
DHE-DSS-AES128-SHA256	DHE	DSS	AES128	SHA256
DHE-RSA-AES128-SHA	DHE	RSA	AES128	SHA
DHE-DSS-AES128-SHA	DHE	DSS	AES128	SHA
DHE-RSA-SEED-SHA	DHE	RSA	SEED	SHA
DHE-DSS-SEED-SHA	DHE	-DSS	SEED	SHA
DHE-RSA-CAMELLIA128-SHA	DHE	RSA	CAMELLIA128	SHA
DHE-DSS-CAMELLIA128-SHA	DHE	DSS	CAMELLIA128	SHA
ECDH-RSA-AES128-GCM-SHA256	ECDH	RSA	AES128-GCM	SHA256
ECDH-ECDSA-AES128-GCM-SHA256	ECDH	ECDSA	AES128-GCM	SHA256
ECDH-RSA-AES128-SHA256	ECDH	RSA	AES128	SHA256
ECDH-ECDSA-AES128-SHA256	ECDH	ECDSA	AES128	SHA256

ECDH-RSA-AES128-SHA	ECDH	RSA	AES128	SHA
ECDH-ECDSA-AES128-SHA	ECDH	ECDSA	AES128	SHA
AES128-GCM-SHA256	RSA	RSA	AES128-GCM	SHA256
AES128-SHA256	RSA	RSA	AES128	SHA256
AES128-SHA	RSA	RSA	AES128	SHA
SEED-SHA	RSA	RSA	SEED	SHA
CAMELLIA128-SHA	RSA	RSA	CAMELLIA128	SHA
PSK-AES128-CBC-SHA	PSK	PSK	AES128-CBC	SHA
ECDHE-RSA-RC4-SHA	ECDHE	RSA	RC4	SHA
ECDHE-ECDSA-RC4-SHA	ECDHE	ECDSA	RC4	SHA
ECDH-RSA-RC4-SHA	ECDH	RSA	RC4	SHA
ECDH-ECDSA-RC4-SHA	ECDH	ECDSA	RC4	SHA
RC4-SHA	RSA	RSA	RC4	SHA
RC4-MDA	RSA	RSA	RC4	MDA
PSK-RC4-SHA	PSK	PSK	RC4	SHA
ECDHE-RSA-DES-CBC3-SHA	ECDHE	RSA	DES-CBC3	SHA
ECDHE-ECDSA-DES-CBC3-SHA	ECDHE	ECDSA	DES-CBC3	SHA
SRP-DSS-3DES-EDE-CBC-SHA	SRP	DSS	3DES-EDE-CBC	SHA
SRP-RSA-3DES-EDE-CBC-SHA	SRP	RSA	3DES-EDE-CBC	SHA
SRP-3DES-EDE-CBC-SHA	SRP	SRP	3DES-EDE-CBC	SHA
EDH-RSA-DES-CBC3-SHA	EDH	RSA	DES-CBC3	SHA
EDH-DSS-DES-CBC3-SHA	EDH	DSS	DES-CBC3	SHA
ECDH-DSS-DES-CBC3-SHA	ECDH	DSS	DES-CBC3	SHA
ECDH-ECDSA-DES-CBC3-SHA	ECDH	ECDSA	DES-CBC3	SHA
DES-CBC3-SHA	RSA	RSA	DES-CBC3	SHA
PSK-3DES-EDE-CBC-SHA	PSK	PSK	3DES-EDE-CBC	SHA
EDH-RSA-DES-CBC-SHA	EDH	RSA	DES-CBC-	SHA
EDH-DSS-DES-CBC-SHA	EDH	DSS	DES-CBC	SHA
DES-CBC-SHA	RSA	RSA	DES-CBC	SHA

Definitions

3DES	Triple-DES
AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
DCM	Dual Counter Mode
DE	Diffe-Hellman
DES	Data Encryption Standard.
DSS	Digital Signature Standard
ECDH	Elliptic Curve Diffe-Hellman
ECDHE	Elliptic Curve Diffe-Hellman Ephemeral
ECDSA	Elliptic Curve Digital Signature Algorithm
EDE	Encrypt-Decrypt-Encrypt

EDH	Ephemeral Diffie-Hellman
GCM	Galois Counter Mode
MAC	Message Authentication Code
PSK	Pre-Shared Key
RC4	Rivest Cyper 4. A symmetric stream cipher.
RSA	Rivent, Shamir and Adelman.
SEED	Block cipher developed by the Korea Information Security Agency (KISA).
SHA	Secure Hash Algorithm
SRP	Secure Remote Password

4.15 Using TN3270 Plus on a Network

TN3270 Plus can be installed on a network disk and accessed by network users. See the License Agreement to review the licensing requirements for network installation.

Network installation is normally done in one of the following ways:

1. The network users all share the same TN3270 Plus configuration settings.
2. The network users use personal TN3270 Plus configuration settings.

TN3270 Plus keeps configuration information in its configuration files.

Shared Configuration and Session Settings

For networks where users will share files, the configuration files should be customized on the network drive. All configuration files should be made read-only. This will allow users to change their session settings, but the changes will not be saved when a session is closed. Optionally, you can lockdown the session settings, so the users cannot change them. The TN3270 Plus Configuration File Folder (Setup, Preferences, Configuration File Folder) should be modified to point to the network drive containing the configuration files. If many of the PCs on the network have multiple users you may want to add:

HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration\DataPath\ (64-bit Windows) or
HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\DataPath\ (32-bit Windows)

to each PC instead of changing the Configuration File Folder for each of the users.

Personal Configuration and Session Settings

For networks where users have their own personal settings, the TN3270 Plus Configuration File Folder (Setup, P references, Configuration File Folder) should be pointed to the user directory containing the configuration files. As an alternative, the Configuration File Folder entry can be deleted and TN3270 Plus started with a working directory, with the working directory pointed at the directory containing the configuration files. If no Configuration File Folder or working directory is specified, TN3270 Plus looks for the configuration files in the install directory (the directory containing the tn3270.exe file). On Windows 8, 7 and Windows Server 2012 and 2008 only, if TN3270 Plus still has not found its data files it will also look in the Windows Application directory. If this directory does not exist, TN3270 Plus will create it and use it for the configuration files.

License Code

When the TN3270 Plus license code is entered, it is stored in the tn3270.ini file. The default location for this file is:

Windows Operating System	tn3270.ini location
11/10/8/7 2022/2019/2016/2012/2008	C:\ProgramData\SDI\TN3270 Plus\
2003/2000/XP	C:\:\Documents and Settings\All Users\Application Data\SDI\TN3270 Plus\

If you wish to store the license code in a different location, see Moving the License Code File.

4.16 Using TN3270 Plus on Windows Terminal Server

TN3270 Plus may be installed on and accessed from a Windows Terminal Server.

Windows Terminal Server installation is normally done in one of the following ways:

1. The Windows Terminal Server clients all share the same TN3270 Plus configuration settings.
2. The Windows Terminal Server clients use personal TN3270 Plus configuration settings.

TN3270 Plus keeps configuration information in its configuration files.

Shared Configuration and Session Settings

For networks where clients will share the same configuration, the configuration files should be customized on the Windows Terminal Server. All configuration files should be made read-only. This will allow clients to change their session settings, but the changes will not be saved when a session is closed. Optionally, you can lockdown the session settings, so the clients cannot change them. The path to the configuration files should be added to the

HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration\DataPath\ (64-bit Windows) or HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\DataPath\ (32-bit Windows)

and propagated to all Windows Terminal Server clients.

Personal Configuration and Session Settings

For networks where clients have their own personal settings, the TN3270 Plus Configuration File Folder (Setup, Preferences, Configuration File Folder) should be pointed to the client directory containing the configuration files. As an alternative the Configuration File Folder entry can be deleted and TN3270 Plus started with a working directory, with the working directory pointed at the directory containing the configuration files.

If no Configuration File Folder or working directory is specified, TN3270 Plus looks for the configuration files in the install directory (the directory containing the tn3270.exe file). On Windows 8, 7, Windows Server 2012, and 2008 only, if TN3270 Plus still has not found its data files it will also look in the Windows Application directory. If this directory does not exist, TN3270 Plus will create it and use it for the configuration files.

License Code

When the TN3270 Plus license code is entered, it is stored in the tn3270.ini file. The default location for this file is:

Windows Operating System	tn3270.ini location
11/10/8/7 2022/2019/2016/2012/2008	C:\ProgramData\SDI\TN3270 Plus\
2003/XP/2000	C:\:\Documents and Settings\All Users\Application Data\SDI\TN3270 Plus\

If you wish to store the license code in a different location, see Moving the License Code File.

4.17 Using TN3270 Plus with Multiple Windows Users

TN3270 Plus can be installed on a PC and accessed by multiple windows users. In this environment, each user can have their own personal set of TN3270 Plus configuration files or the TN3270 Plus configuration files can be shared by all users. By default each user has their own set of configuration files.

Personal Configuration and Session Settings

With the default installation, the TN3270 Plus configuration files are located in the following directory:

Windows Operating System	Default Configuration Files Folder Path

11/10/8/7 2022/2019/2016/2012/2008	C:\Users*user_id*\AppData\Roaming\SDI\TN3270 Plus\
2003/ XP/2000	C:\Program Files\SDI\TN3270 Plus\

Shared Configuration and Session Settings

If you want multiple users on the same PC to share the same TN3270 Plus configuration files you need to change the configuration file location to a folder that is shared by all users. You can do this by logging onto each user and changing the Configuration Files Folder (Setup, Preferences, Configuration Files Folder) to the shared folder or you can delete the Configuration Files Folder and specify the shared folder in the following registry key:

HKEY_LOCAL_MACHINE\Software\Wow6432Node\SDI\TN3270 Plus\Configuration\DataPath\ (64-bit Windows)HKEY_LOCAL_MACHINE\Software\SDI\TN3270 Plus\Configuration\DataPath\ (32-bit Windows)

4.18 Deployment

4.18.1 Silent or Unattended Uninstall

If you plan to uninstall TN3270 Plus from a large number of workstations, you may want to automate the uninstall. A silent or unattended uninstall allows you to uninstall TN3270 Plus using a single command from the Windows command line.

Silent Uninstall Using TN3270PlusSetup.exe

Use one of the following commands to install TN3270 Plus.

`TN3270PlusSetup.exe /x /v/qn`	Installs TN3270 Plus for the current user.
`TN3270PlusSetup.exe /x /v"/qn ALLUSERS=1"`	Installs TN3270 Plus for all users.

The TN3270PlusSetup.exe file must be the same release level as the installed version of TN3270 Plus or the uninstall command will not work. For example, the TN3270PlusSetup.exe file for release 4.0.5 cannot be used to uninstall TN3270 Plus release 3.7.5,

Silent Uninstall Using msiexec.exe

If you do not have the TN3270PlusSetup.exe file you can use the msiexec.exe program. Copy the command from the Uninstall TN3270 Plus shortcut and modify it to do a silent uninstall.
To copy the command, right click on the Uninstall TN3270 Plus icon in the Start menu and select Properties. In the Properties dialog box, click on the shortcut tab, copy the command line from the Target Edit box.

You will have a command that looks like this:

```
C:\Windows\SysWOW64\msiexec.exe /x {guid}
```

Where:

guid - is a release dependent set of numbers that identifies the version of TN3270 Plus.

Insert " /qn " into the command prior to "/x" so your command looks like this:

```
C:\Windows\SysWOW64\msiexec.exe /qn /x {guid}
```

Here is an example for release 3.7:

```
C:\Windows\SysWOW64\msiexec.exe /qn /x {D3D7C70C-2EE3-4788-986E-0B93008F3CCB}
```

Execute the command at a Windows command prompt running with Administrator privileges.

4.18.2 Silent or Unattended Install

If you plan to install TN3270 Plus on a large number of workstations, you may want to automate the installation. A silent or unattended install allows you to install TN3270 Plus using a single command from the Windows command line.

Silent Install

Use one of the following commands to install TN3270 Plus.

`TN3270PlusSetup.exe /s /v/qn`	Installs TN3270 Plus for the current user.
`TN3270PlusSetup.exe /s /v"/qn ALLUSERS=1"`	Installs TN3270 Plus for all users.
`TN3270PlusSetup /s /v"/qn INSTALLDIR=\"f:\My Test\""`	Installs TN3270 Plus in the specified directory

The silent install uses the following default settings:

Application Folder	c:\Program Files\SDI\TN3270 Plus c:\Program Files (x86)\SDI\TN3270 Plus (64-bit)
Application Data Folder	**Windows 11/10/8/7 Windows Server 2022/2019/2016/2012/2008:** C:\Users*user_name*\AppData\Roaming\SDI\TN3270 Plus\ **Windows 2003/XP/2000**: C:\Documents and Settings*user_name*\Application Data\SDI\TN3270 Plus\

Adding Configuration Files

If you want to pre-configure settings for your users, you will need to copy TN3270 Plus configuration files into the user's Application Data Folder. We recommend you set up one TN3270 Plus workstation with all the desired settings and then copy the configuration files from that workstation to your other users' workstations.

Adding Preferences

The steps above are all that is needed for most installations. However, if you want to include TN3270 Plus preferences (Setup, Preferences) in your pre-configured system, you need to add TN3270 Plus registry entries to each user's system.

Export the TN3270 Plus Registry entries from your pre-configured workstation. To export the TN3270 Plus Registry entries, run "regedit.exe". In the **Registry Editor,** select the following folder:

```
HKEY_CURRENT_USER\Software\SDI\TN3270 Plus
```

Open the **Registry** menu and select **Export Registry File...** Save the exported file as "tn3270.reg". The exported file can be imported to the receiving system by the following command. The "-s" switch suppresses information dialog boxes.

```
regedit -s c:\path\tn3270.reg
```

4.18.3 Extracting the .MSI File

If you need an .msi (Microsoft Installer) file to deploy TN3270 Plus, you can extract the .msi file from the TN3270 Plus distribution file.

To extract the TN3270 Plus .msi file:

1. Create a directory for the .msi file.
2. Open a Windows command prompt and run the TN3270 Plus distribution file with the "/a" option. The " /a" option performs an administrative installation. It copies the .msi file, the programs files and the application data files from the TN3270 Plus distribution file into the specified directory. For example:

   ```
   TN3270PlusSetup.exe /a
   ```

3. Follow the install wizard instructions. You will be instructed to specify a network folder. This can be any folder. When the install wizard completes this folder will contain the following:

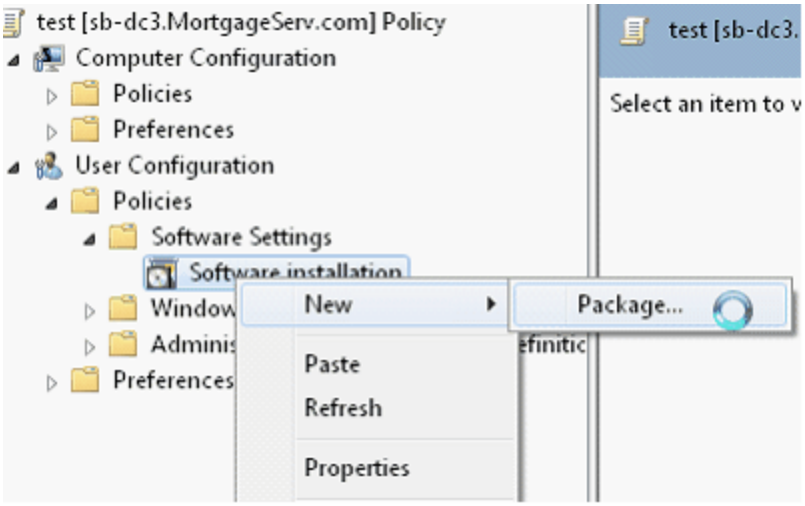
   tn3270*vmm*.msi
   Application Data folder
   Program Files folder

4.18.4 Using Group Policy to Deploy TN3270 Plus

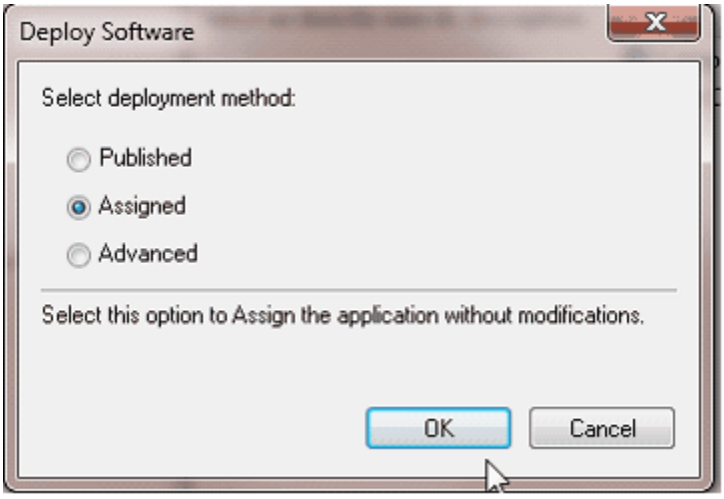
You can use Group Policy to deploy TN3270 Plus to your users. You can use Group Policy to assign or publish software to users or computers in a domain.

This topics outlines the steps required to deploy TN3270 Plus using Group Policy. Use these steps as a guideline. You may need to modify the steps slightly to meet your company's requirements.

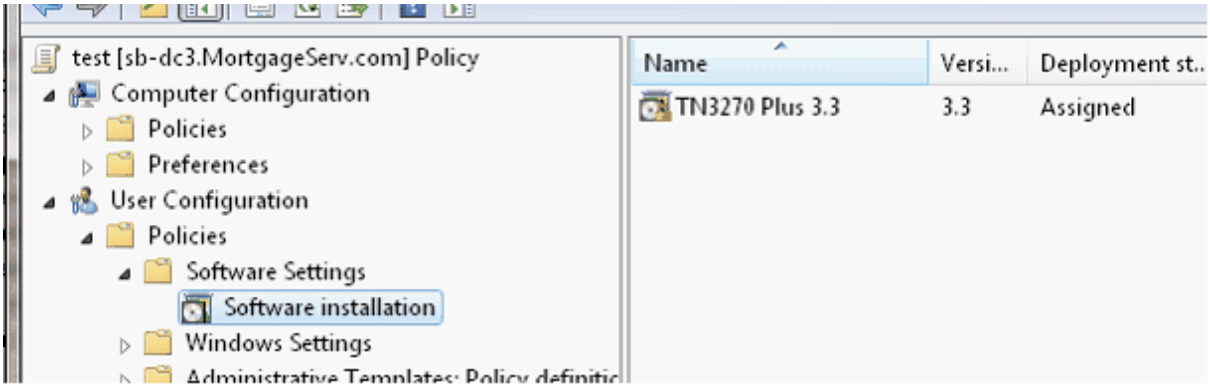
1. Extract the TN3270 Plus .MSI (Microsoft Installer) file from the TN3270 Plus distribution file. See Extracting the .MSI File for instructions.
2. Copy the extracted MSI file to shared folder accessible to read and execute by your users.
3. Open the Group Policy Editor. (Click Start, click Run..., type gpedit.msc and click OK)
4. Navigate to "Software Installation".
5. Right click on "Software Installation" and select **New** and then **Package...** from the context menu.



6. Browse to the .MSI file saved in Step 2.
7. Set the deployment method.



8. The package now shows in Software Installation.



9. The software package will be deployed to Group members. How and when the package is deployed depends on the deployment method you chose. See the descriptions of the different deployment methods below.

Deployment Methods

Method	Description
Published	Publishing an application doesn't actually install the application, but rather makes it available to users. Publishing is a group policy setting, so it will not take effect until the next time the user logs in. When the user does log in, he or she will not initially notice anything different. However, if the user were to open the Control Panel and click on the Add / Remove Programs option, they will find that TN3270 Plus is now on the list. A user can then choose to install TN3270 Plus on their machine.
Assigned	Assigning is a group policy action, so the assignment won't take effect until the next time that the user logs in. When the user does log in, he or she will see that the new application has been added to the Start menu and/or to the desktop. Although a menu option or an icon for the application exists, the software hasn't actually been installed. To avoid overwhelming the server containing the installation package, the software is not actually installed until the user attempts to use it for the first time.

Advanced	Used the advanced deployment option if you need to publish or assign TN3270 Plus with transforms.

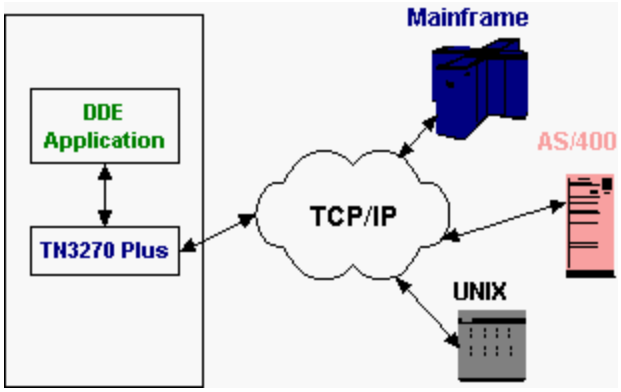
4.19 Using DDE Support

4.19.1 DDE Overview

The TN3270 Plus DDE support allows you to write Windows applications that communicate with IBM mainframe, iSeries (AS/400) or UNIX computers via TN3270 Plus terminal emulation.

How DDE Support Works

TN3270 Plus can act as a DDE server that fulfills requests of your application, the DDE client. This allows your application to send data to, or receive data from, the host computer via TN3270 Plus. TN3270 Plus can also act as a DDE client and initiate data exchanges with a DDE application server. TN3270 Plus DDE script commands can initiate a DDE conversation and exchange data with other DDE applications. DDE is most commonly used to exchange data with Microsoft Word or Excel, but it can be used with any application that supports DDE.



Sample Programs

Sample programs are available for download from the Customer Downloads page (http://sdisw.com/customer_downloads.htm) on our web site.

See Also:

DDE Functions
DDE Keystroke Table

4.20 Using WinHLLAPI Support

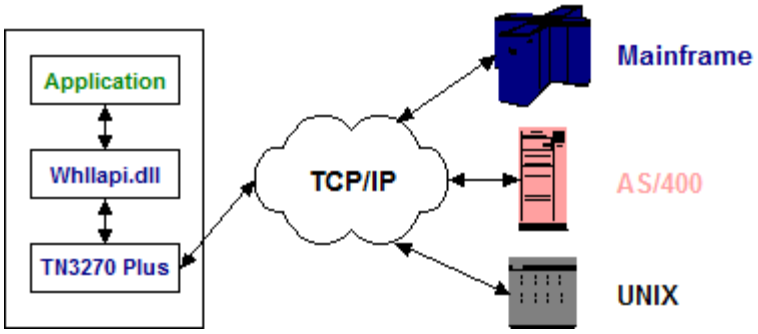
4.20.1 WinHLLAPI Overview

The TN3270 Plus WinHLLAPI support allows you to write Windows applications that communicate with midrange or mainframe computers via TN3270 Plus terminal emulation.

TN3270 Plus WinHLLAPI supports most of the functions in the Windows HLLAPI Version 1.1 specification. See WinHLLAPI Unsupported Functions for a list of functions that TN3270 Plus does not support.

How WinHLLAPI Support Works

Your application needs to load the Whllapi.dll (32-bit) or Whllapi64.dll (64-bit) distributed with TN3270 Plus. Once this is done, your application may call the Whllapi functions to send data to, or receive data from, the host computer. Whllapi.dll and Whllapi64.dll are extensions of TN3270 Plus and TN3270 Plus to send data to, or receive data from, the host computer.



4.20.2 WinHLLAPI Getting Started

You need the following to use TN3270 Plus WinHLLAPI support:

1. Your application needs to load the Whllapi.dll or Whllapi64.dll distributed with TN3270 Plus
2. You must create a TN3270 Plus session that includes a "WinHLLAPI Session ID" (Setup, Sessions…, Setup Items = Host, WinHLLAPI Session ID). This session must be connected and your application must specify the session ID.

WinHLLAPI Technical Specification

You may download the WinHLLAPI 1.1 technical specification from our web site. This document describes all the WinHLLAPI functions.

http://sdisw.com/download/winhllapi_specification.zip

WinHLLAPI Sample Programs

Sample programs are available for download from the Customer Downloads page on our web site.

WinHLLAPI Keystroke Table

Special mnemonics are used to send function and AID keys. For example, @E is used to send the Enter key. The WinHLLAPI keystroke table uses the same mnemonics as the DDE Keystroke Table documented in this help file.

4.20.3 WinHLLAPI Unsupported Functions

The TN3270 Plus WinHLLAPI support does not currently support the following functions. If you need one of these functions please send an e-mail to support@sdisw.com

WinHLLAPIAsync()
WinHLLAPIIsBlocking()
WinHLLAPICancelAsyncRequest()
WinHLLAPICancelBlockingCall()
WinHLLAPISetBlockingHook()
WinHLLAPIUnhookBlockingHook()

5 Menus

5.1 Host Menu

5.1.1 Command Summary

The Host menu contains the following commands:

Connect...	Connects to a host computer.
Close	Closes an open connection.
Save	Saves the screen image to the active log file.
Save As...	Saves the screen image to the selected log file.
Print Screen	Prints the terminal screen.
Print Setup...	Selects a printer and a printer connection.
Print Preview	Shows how the terminal screen will look when it is printed.
Close PrintScreen File	Close the Print Screen spool file and send any print screens held in the Print Screen file to the printer.
File Transfer...	Transfers a file to or from the host computer.
Cancel File Transfer	Cancels an active file transfer.
Run Script...	Runs a script.
Edit Script...	Edit a script.
Cancel Script	Cancels an active script.
Enable Logging	Start writing terminal screen activity to the session log file.
Disable Logging	Stop writing terminal screen activity to the session log file.
Exit	Exits TN3270 Plus.

5.1.2 Connect command

Connects you to a host computer. Specify the host computer in the Connect to Host dialog box.

System Command:

HostConnect

Shortcuts:

Standard Toolbar 

5.1.3 Close command

Use this command to close (disconnect) the current session.

A message box appears asking you to confirm that you want to close the session if "Ask for confirmation if I close a session" is set on in the TN3270 Plus Preferences (Setup, Preferences...).

System Command:

HostClose

Shortcuts:

Standard Toolbar: 

5.1.4 Save command

Save the screen image to the session **Log File**. If no **Log File** is defined, the **Save As** dialog box prompts you for a file name. Define the **Log File** for a session in the Host pane of the **Session Setup** dialog box.

System Command:

HostSave

Shortcuts:

Standard Toolbar: 

5.1.5 Save As command

Save the screen image to a user specified log file. This command displays a dialog box allowing you to select the log file to receive the screen image or to enter a new log file name. The new log file remains active until the **Save As...** command is used to activate another log file or the session ends.

System Command:

HostSaveAs

5.1.6 Print Screen command

Print the screen image on the printer. Use the Print Preview command to see what the screen image will look like when you print it. Use the Print Setup command to change your printer settings.

System Command:

HostPrintScreen

Shortcuts:

Standard Toolbar: 

Keys: Ctrl+P

5.1.7 Print Setup command

Use this command to select a printer and a printer connection. This command presents the Print Setup dialog box where you specify the printer and its connection.

System Command:

HostPrintSetup

5.1.8 Print Preview command

Use this command to display the current TN3270 Plus session screen as it would appear when printed. When you choose this command, the main window is replaced with a print preview window in which one or two pages are displayed in their printed format. The print preview toolbar offers you options to view either one or two pages at a time; move back and forth through the document; zoom in and out of pages; and initiate a print job.

System Command:

HostPrintPreview

5.1.9 Close PrintScreen File command

Use this command to print Print Screen images in the Print Screen spool file. The Print Screens to Spool Before Printing option allows multiple print screens to be printed on a single page. Print screen output is held in a print spool file until the specified number of print screens is reached, then the print screen output is sent to the printer. Issue the Close PrintScreen File command to send print screen output held in the print spool file to the printer before the specified number of print screens is reached. The Print Screens to Spool Before Printing option and the number of print screens to spool are specified in the Printer pane of the Session Setup dialog box.

When the Print Screens to Spool Before Printing option is active and print screen images are held in the spool file, a printer icon followed by the number of print screens held in the print spool file appears on the status bar.

System Command:

HostClosePrintScreenFile

5.1.10 File Transfer command

3270 Terminal Sessions

Use this command to transfer a file from the host computer to your PC or from your PC to the host computer. Specify the file you want to transfer in the File Transfer dialog box.

It is also possible to transfer a file using a script. If you often transfer the same file, a script is a more automated way to perform the operation. See the FileTransfer script command for more information on file transfer operations using a script.

5250, VTxxx and ANSI Terminal Sessions

Use this to command to start SDI FTP. This command is disabled if the FTP feature is not licensed.

System Command:

HostFileTransfer

Shortcuts:

Standard Toolbar: 

5.1.11 Cancel File Transfer command

Use this command to cancel a file transfer in progress. A message box will appear asking you to confirm the cancellation of the file transfer.

System Command:

HostCancelFileTransfer

5.1.12 Start SDI FTP

Use this command to start SDI FTP.

System Command:

HostStartSdiftp

5.1.13 Run Script command

Use this command to run a script. You may choose from the list of recently used scripts. Or, click Browse… to display the Select Script File dialog box, click on the desired script file and then click the Open button to run the script.

System Command:

HostRunScript

Shortcuts:

Standard Toolbar: 

5.1.14 Edit Script command

Use this command to edit a script with Notepad. You may choose from the list of recently used scripts. Or, click **Browse…** to display the **Select Script File** dialog box, click on the desired script file and then click the **Open** button to edit the script.

5.1.15 Cancel Script command

Use this command to cancel an active script. A message box will appear asking you to confirm the cancellation of the script.

System Command:

HostCancelScript

5.1.16 Enable Logging command

Use this command to start logging screen activity to the session log file. Logging screen activity to a disk file gives you a record of your session. The session log file is defined in the host pane of the Session Setup dialog (Setup, Sessions…, Setup Items = Host, Log File edit box). The session log file may be edited with any text editor.

If you always want logging active for a session, set the "Enable logging" option for the session (Setup, Sessions…, Setup Items = Host, Enable logging check box).

System Command:

HostEnableLogging

Shortcuts:

Standard Toolbar:  (Use the Customize Toolbar dialog box. to add this button.)

5.1.17 Disable Logging command

Use this command to stop logging screen activity to the session log file. This command is not displayed unless session logging has already been started by the Enable Logging command (Host, Enable Logging) or the "Enable logging" option (Setup, Sessions, Setup Items = Host, Enable logging check box).

System Command:

HostDisableLogging

Shortcuts:

Standard Toolbar:  (Use the Customize Toolbar dialog box. to add this button.)

5.1.18 Host 1, 2, 3, 4, 5, 6, 7, 8, 9 command

Use the numbers and session names listed at the bottom of the Host menu to switch to another active session. Choose the number that corresponds to the desired session.

System Command:

HostSession1
HostSession2
HostSession3
…

Shortcuts:

Standard Toolbar:  Click on the toolbar button of the desired session.

Keys: Alt+n Where n is the session number you wish to switch to.

5.1.19 Exit command

Terminate TN3270 Plus. You can also use the Close (Alt+F4) command on the application control menu. TN3270 Plus prompts you to close all connected sessions before terminating.

System Command:

HostExit

Shortcuts:

Mouse: Double-click the application's Control menu button.

Keys: ALT+F4

5.2 Edit Menu

5.2.1 Command Summary

The Edit menu contains the following commands:

Undo	Undo the last data entry.
Redo	Reverse an undo operation.
Cut	Delete the selected data from the screen and move it to the clipboard.
Cut Append	Delete the selected data from the screen and append it to the data already on the clipboard.
Copy Options	Select the mode of operation for the copy menu item.
Copy	Copy the selected data from the screen to the clipboard as text.
Copy Append	Copy the selected data from the screen and append it to the data already on the clipboard.
Copy As Image	Copy the selected data from the screen to the clipboard as a bitmap image.

Paste Mode	Select the mode of operation for the paste menu item.
Paste	Inserts data from the clipboard onto the screen.
Paste Continue	Continue pasting from where the last paste operation completed.
Selection Mode	Change the selection display mode.
Select All	Selects all the data on the screen.
Deselect	Cancels the current selection.
Print Clipboard	Prints the current contents of the clipboard.
Save Clipboard	Save the current contents of the clipboard to a disk file.
Clear Clipboard	Save the current contents of the clipboard to a disk file.

5.2.2 Undo command

Use this command to undo the last data entry made on the terminal screen. You may use the Undo command multiple times to remove multiple data entries. Undo points are cleared each time the terminal emulation screen is updated by the host computer. Use the Redo command to reverse an undo operation.

The Undo command is supported for 3270 and 5250 terminal sessions.

System Command:

EditUndo

Shortcuts:

Keys: Ctrl+Z

5.2.3 Redo command

Use this command to reverse the last undo operation. You may use the Redo command multiple times to reverse multiple undo operations.

The Redo command is supported for 3270 and 5250 terminal sessions.

System Command:

EditRedo

Shortcuts:

Keys: Ctrl+Y

5.2.4 Cut command

Use this command to place the currently selected data on the clipboard in text and RTF fomat. Unprotected data is replaced with blanks and protected data remains on the screen. This command is unavailable if no data is currently selected.

Data that you place on the clipboard remains there until you replace it with a new item.

System Command:

EditCut

Shortcuts:

Standard Toolbar: 

Keys: Ctrl+X

5.2.5 Cut Append command

Use this command to append the currently selected data to the clipboard in text and RTF formats. Unprotected data is replaced with blanks and protected data remains on the screen. This command is unavailable if no data is currently selected.

Data that you place on the Clipboard remains there until you replace it with a new item.

System Command:

EditCutAppend

Shortcuts:

none

5.2.6 Copy Options command

Use this command to select the mode of operation for the copy command:

Change Field Attribute to Tab	Replaces field attributes with tabs when data is copied to the clipboard. The text on the clipboard is then tab delimited. This option allows columns of data separated by field attributes to be pasted into multiple columns in a spreadsheet program. (For example, Microsoft Excel.) (3270 and 5250 sessions only)
Change Spaces to Tab	Changes two or more contiguous spaces into a single tab character when data is copied to the clipboard. This allows data in a single field that is separated by spaces to be pasted into multiple columns of a spreadsheet program. (For example, Microsoft Excel.) (3270 and 5250 sessions only)
Prefix Copy Append with Newline	Inserts a newline (cr/lf) between existing clipboard data and the new clipboard data during a Copy Append operation.
Auto Copy After Select	Automatically copies the selected text to the clipboard when the left mouse button is released and text has been selected.
Remove Trailing Spaces	Removes trailing spaces from selected data before it is copied to the clipboard.
Copy All if No Selection	Copies all the data in the session display area if the copy command is issued with no selection active.

System Command:

EditCopy

Shortcuts:

none

5.2.7 Copy command

Use this command to copy selected screen contents onto the clipboard in text and RTF formats. This command is unavailable if no data is currently selected.

System Command:

EditCopy

Shortcuts:

Standard Toolbar: 

Keys: Ctrl+C

5.2.8 Copy Append command

Use this command to copy the currently selected data and append it to data already on the clipboard. This command is unavailable if no data is currently selected.

System Command:

EditCopyAppend

Shortcuts:

none

5.2.9 Copy As Image

Use this command to copy the selected screen area onto the clipboard as a bitmap image. If there is no selection, the entire terminal window is copied to the clipboard. Use this command to capture screen shots of all or part of the terminal window.

System Command:

EditCopyAsImage

Shortcuts:

none

5.2.10 Paste Mode command

Use this command to select the mode of operation for the paste menu item. The following options are available:

Block Mode	Pastes line by line. Each line is pasted starting at the current cursor column. Pasting continues until the end of the unprotected field. Use this mode when pasting blocks of data into a text editor. (3270 and 5250 sessions only)
Block Mode with Word-Wrap	Works like block mode except it breaks lines of data at a word boundary so that words are not split across lines. Use this mode when pasting blocks of data into a text editor. (3270 and 5250 sessions only)
Overlay Block Mode	Pastes line by line. Each line is pasted starting at the current cursor column and continues until the end of the line. The clipboard line overlays the new line character-for-character filling in all unprotected fields; protected fields are bypassed as are the associated characters on the clipboard. Use this mode when pasting a block that contains data from lines containing multiple fields (for example, a data entry screen). (3270 and 5250 sessions only)
Stream Mode	Pastes character by character into unprotected fields. When a protected field is encountered the paste continues with the next clipboard character in the next unprotected field. When all unprotected fields on a line are filled, pasting moves to the first unprotected field on the next line. Pasting stops when all characters have been pasted or the paste operation reaches the end of the screen. (3270 and 5250 sessions only)
Move Cursor After Paste	Move the cursor to the end of the pasted data after a paste operation instead of leaving it at the original location. (3270 and 5250 sessions only)
Replace Tab with Space	Replaces each tab character with a space when the characters are pasted. (3270 and 5250 sessions only)
Erase EOF After Paste	Issue an Erase EOF command to erase the contents of the field that follows the pasted text. (3270 and 5250 sessions only)

Shortcuts:

none

5.2.11 Paste Continue command

Use this command to continue the paste operation where the last paste operation completed. This command is unavailable until a paste operation completes without pasting the entire contents of the clipboard.

For example, use paste continue to paste a large quantity of data into a host edit session. A paste operation ends when it reaches the end of the terminal screen. If the clipboard still contains more data, you may move to the next terminal screen in your edit session and use Paste Continue to start pasting from the clipboard where the prior paste or paste continue operation ended.

System Command:

EditPasteContinue

Shortcuts:

none

5.2.12 Paste command

Use this command to insert a copy of the clipboard contents at the current cursor location. This command is unavailable if the clipboard is empty.

System Command:

EditPaste

Shortcuts:

Standard Toolbar: 

Keys: Ctrl+V

5.2.13 Selection Mode command

Use this command to choose the selection mode for cut, copy and paste operations.

Reverse Video	displays selected data in reverse video.
Rubber Band	displays selected data in a rectangle containing diagonal hash lines.

Shortcuts:

Keys: none

5.2.14 Select All command

Use this command to select all the data in the session display area.

System Command:

EditSelectAll

Shortcuts:

Keys: Ctrl+A

5.2.15 Deselect command

Use this command to cancel a data selection.

System Command:

EditDeselect

Shortcuts:

Keys: Ctrl+D

5.2.16 Print Clipboard command

Use this command to print the contents of the clipboard. This command is only available when the clipboard contains text. It is disabled when the clipboard contains something other than text (for example, a bit map).

System Command:

EditPrintClipboard

Shortcuts:

none

5.2.17 Save Clipboard command

Use this command to save the contents of the clipboard to the session Clipboard File. If no session Clipboard File is defined, the Save As dialog box prompts you for a file name. Define the Clipboard File for a session in the Host pane of the Session Setup dialog box.

This command is only available when the clipboard contains text. It is disabled when the clipboard contains something other than text (for example, a bit map).

System Command:

EditSaveClipboard

Shortcuts:

none

5.2.18 Clear Clipboard command

Use this command to clear the contents of the clipboard.

System Command:

EditClearClipboard

Shortcuts:

none

5.3 View Menu

5.3.1 Command Summary

The View menu contains the following commands:

Standard Toolbar	Shows or hides the standard toolbar.
Keypad Toolbar	Shows or hides the keypad toolbar.
Status Bar	Shows or hides the status bar.
Operator Information Area	Shows or hides the Operator Information Area.
Full Screen	Toggles window mode and full screen mode display.
Response Time...	Displays the response time frequency distribution dialog box.
Next Session	Displays the next active session.
Previous Session	Displays the previous active session.

5.3.2 Standard Toolbar command

Display or hide the Standard Toolbar. This toolbar includes buttons for some of the most common commands in TN3270 Plus. A check mark appears prior to the menu item when the toolbar is displayed.

See the Standard Toolbar section for help on using the standard toolbar.

System Command:

ViewStandardToolbar

5.3.3 Keypad Toolbar command

Display or hide the keypad toolbar. This toolbar includes buttons for terminal keys. You may use these buttons in place of the keyboard keys. A check mark appears prior to the menu item when the toolbar is displayed.

System Command:

ViewKeypadToolbar

5.3.4 Status Bar command

Display or hide the Status Bar. The Status Bar shows messages and information that describe the state of your terminal emulation session. A check mark appears prior to the menu item when the Status Bar is displayed.

See the Status Bar section for complete descriptions of the fields on the status bar.

System Command:

ViewStatusBar

5.3.5 Operator Information Area command

Display or hide the 3270 Operator Information Area (OIA). The Operator Information Area is a single line at the bottom of the terminal emulation display that describes the state of your terminal emulation session. A check mark appears prior to the menu item when the Operator Information Area line is displayed.

System Command:

ViewOperatorInformationArea

5.3.6 Full Screen command

Change the display from window mode to full screen mode or from full screen mode to window mode.

Note: When switching to full screen mode TN3270 Plus selects the font size that best fits the full screen. If the display font is not resizable (for example, fixedsys), the font size will not change when you switch to full screen mode.

System Commands:

ViewFullScreen	Toggles full screen mode
ViewFullScreeOn	Activates full screen mode
ViewFullScreenOff	Ends full screen mode

Shortcuts:

Standard Toolbar: 

5.3.7 Response Time command

Displays the **Response Time Frequency Distribution** dialog box. The **Response Time Frequency Distribution** dialog box is a summary of the response times for all active sessions. Use this information to monitor the performance of your Internet connection.

The following is a summary of the fields in the **Response Time Frequency Distribution** dialog box:

Heading	Description
Session	The session number followed by the session description.
< 1	Number of subsecond responses.
1 < 2	Number of responses greater than or equal to 1, but less than 2 seconds.
2 < 3	Number of responses greater than or equal to 2, but less than 3 seconds.
3 < 5	Number of responses greater than or equal to 3, but less than 5 seconds.
5 < 7	Number of responses greater than or equal to 5, but less than 7 seconds.
7 < 10	Number of responses greater than or equal to 7, but less than 10 seconds.
10+	Number of responses greater than or equal to 10 seconds.
Average	Average response time while the session was active.
Highest	Highest response time while the session was active.

System Command:

ViewResponseTime

5.3.8 Next Session command

Changes the display to the next active session.

System Command:

ViewNextSession

Shortcuts:

Keys: Ctrl+Tab

5.3.9 Previous Session command

Changes the display to the previous active session (for example, from session 3 to session 2.)

System Command:

ViewPreviousSession

Shortcuts:

Keys: Shift+Ctrl+Tab

5.4 Setup Menu

5.4.1 Command Summary

The Setup menu contains the following commands:

Sessions...	Tailor a terminal emulation session.
Host...	Change connection settings.
Terminal...	Change the terminal type and terminal options.
Colors...	Change the session colors.
Display...	Change the session display options, for example, screen font, cursor style, window attributes.
Printer...	Change printer options.
Keyboard...	Change the keyboard configuration.

Toolbars...	Configure the TN3270 Plus toolbars.
Proxy Server...	Configure to connect to a host computer via a proxy server.
Security...	Configure security (TLS or SSH) options.
Preferences...	Change application behavior.
Desktop Shortcut....	Create a shortcut for starting TN3270 Plus on your desktop.
Trace Socket...	Turns TCP/IP socket tracing on or off.

5.4.2 Session Command

The **Session Setup** dialog box contains all the customization options for your terminal emulation sessions.

Step by Step Instructions for Using the Session Setup Dialog Box.

1. Specify a session name in the **Session Name** drop-down list box. You may select a session from the drop-down list or create a new session name using the **Save As...** button. If the **Save As...** button is used to create a new session then the new session inherits all the properties of the selected session (except the session name of course).
2. Select a category from the **Setup Items** list. The pane on the right-hand side of the dialog box displays the options for that category. Make any changes you desire and repeat the process for each category you wish to tailor. Links to help information for each of the categories are listed below.
3. Click on the **Cancel** button to exit the dialog box without saving your changes. Click on the **Apply** button to save your changes, but remain in the dialog box. Click on the **OK** button to save your changes and exit the dialog box.

Panes in the Session Setup Dialog.

Each of the categories in the **Setup Items** list box has an associated pane that appears on the right-hand side of the **Session Setup** dialog box. The following list contains links to help information for each of the panes.

Link to Help for Setup Items:

Host	Specify the host connection options.
Terminal	Define the terminal type.
Colors	Specify the color scheme for your terminal emulation session.
Display	Specify the screen font, cursor style and the terminal window attributes.
Printer	Specify the printer font and printer options.
Keyboard	Configure your keyboard.
Toolbars	Configure your toolbars.
Proxy Server	Configure to connect to a host computer via a proxy server.
Security	Specify SSL or SSH options for a secure connection.

System Command:

SetupSessions

5.4.3 Host command

Use this command to change the connection options for the current terminal emulation session. This command displays the Host pane of the Session Setup dialog box. The Host pane has the following options.

Host Name	Enter the host name of the host computer. You may specify multiple host names separated by commas (e.g. 127.0.0,1,127.0.0.2) If connection to the first host name in the list fails, TN3270 Plus attempts to connect to the second host name in the list and so on throughout the list.
Telnet Port	Enter the telnet connection port. This is normally **23**.

Code Page

(Optional) Enter a host language code page (*.cpg) file. The code page determines the character translations for the keyboard. SDI supplies code pages for many countries (for example, GermAust.cpg and France.cpg). Click the **Browse...** button to display your current code page in the Code Page Setup dialog box.

To select a new code page from the **Code Page Setup** dialog box, select a code page from the drop-down list box or click the **Browse...** button to display the **Select Code Page File** dialog box and select a new *.cpg file. If no code page is specified, the default code page is used (Setup, Preferences...).

Script File

(Optional) Enter a script file name (*.txt). The script begins running as soon as your connection is complete. Use this script to automate your logon. Click the **Browse...** button to display the **Select Script File Name** dialog box.

Log File

(Optional) Enter a log file name (*.log). The default log file name is your session name appended with ".log." Click the **Browse...** button to display the **Select Log File Name** dialog box. You may include one or more of the following variables anywhere in the file name and the variable will be replaced with the associated data.

Variable	Case	Substitution
$COMPUTERNAME	(upper case)	Computer name.
$DATE	(upper case)	Current date in YYYY-MM-DD format.
$SESSION	(upper case)	Current session name.
$USERNAME	(upper case)	Current logged on user name.

For example, a log file name of "$DATE Mainframe.log" creates a new log file each day named "YYYY-MM-DD Mainframe.log". You may also specify Windows environmental variables (e.g. %computername%, %username%, %appdata%, %programfiles%, etc). Check the "Enable logging" check box on this pane to set session activity logging on for this session.

The following commands also use the log file:
Host, Save saves a screen images to the log file.
Host, Enable Logging starts session activity logging.

Clipboard File

(Optional) Enter a file name (*.txt) for saving clipboard text. You may include one or more of the following variables anywhere in the file name and the variable will be replaced with the associated data.

Variable	Case	Substitution
$COMPUTERNAME	(upper case)	Computer name.
$DATE	(upper case)	Current date in YYYY-MM-DD format.
$SESSION	(upper case)	Current session name.
$USERNAME	(upper case)	Current logged on user name.

For example, a clipboard file name of "$DATE Clipboard.txt" creates a new clipboard file each day named "YYYY-MM-DD Clipboard.txt". You may also specify Windows environmental variables (e.g. %computername%, %username%, %appdata%, %programfiles%, etc). The SaveClipboard... command (**Edit**, **Save Clipboard...**) saves clipboard text to this file.

Preferred Session Number

(Optional) Enter a preferred session number (1-9) for this session. If the session number is available, the session will connect using that session number. If the session number is not available, the session will connect using the first available session number.

Keep-alive Interval

(Optional) Specify a keep-alive interval. When set, TN3270 Plus sends a telnet NOP (or an SSH keep-alive if SSH is being used) at the specified interval. The purpose of this option is to simulate activity to keep inactive sessions from being dropped by the host computer. If the keep-alive setting does not work, you may wish to send something other than a telnet NOP (or an SSH keep-alive) at the specified interval, see Changing the Keep-alive

String.

Connect at startup Check this item to automatically connect this session each time TN3270 Plus is started.

Reconnect if disconnected by host Check this item to have TN3270 Plus reconnect your session if it is disconnected by the host computer.

Delay ?? seconds Enter the number of seconds TN3270 Plus should wait before attempting to reconnect a disconnected session.

Bring to top when host updates screen Check this item if you wish TN3270 Plus to switch to this session each time the terminal emulation screen is updated by the host computer.

Start in a new window Check this item if you wish TN3270 Plus to start this session in its own window. When this option is checked, a new window is created if the existing window already contains an active session. If this option is not checked, the session is started as an additional session in the existing TN3270 Plus window. If you want all of your sessions started in new windows, check the "Start each session in a new window" box in the **Preferences** dialog box. (Setup, Preferences, Start each session in a new window)

Suppress alarms from host Check this item to suppress the "beep" when the host sends an alarm bit (3270/5250) or a "bell" control character (VTxxx/ANSI).

Enable logging Check this item to log screen activity to the log file defined in the log file edit box. This log file is cleared each time the session is connected. To prevent the log file from being cleared, check the "Append to existing log file" option.

Append to existing log file Check this item to append data to the exiting log file rather than clearing the log file when the session is connected.

WinHLLAPI Session ID (Optional) Select a short name session id for this session. Choose a letter from A to Z. A short name session id is required for another application to use WinHLLAPI to communication with the session.

Changing the Keep-alive String

In rare instances, the default keep-alive string setting may not work with your host computer. In this case, you may change the keep-alive command string. By default, TN3270 Plus sends hexadecimal FFF1 at each keep-alive interval. To change the default, add the KeepAliveString parameter to the end of the session profile file:

```
KeepAliveString=xxxx
```

Where:

xxxx is the hexadecimal string to send or a terminal key. For example, FFF6 will send IAC, AYT (AYT = Are You There) instead of FFF1 (IAC, NOP (NOP = No Operation)). PF12 will send the PF12 key.

Examples:

```
KeepAliveString=FFF6
KeepAliveString=PF12
KeepAliveString=PF24
KeepAliveString=F12
```

Use caution when specifying a terminal key, so that if the keep-alive interval expires and sends that key it does not cause an unintended result in your application.

The session profile file name is the session name with the .tsp extension. For example, if your session name is "IBM Mainframe" the corresponding session profile name is "IBM Mainframe.tsp." To change the keep-alive string to FFF6:

1. Edit the session profile with notepad.exe or another word processor.
2. Add the following line to the end of the session profile:

   KeepAliveString=FFF6
3. Save the updated session profile and restart TN3270 Plus.

5.4.4 Terminal command

Use this command to change the terminal type and terminal options for the current terminal emulation session. This command displays the Terminal pane of the **Session Setup** dialog box. The Terminal pane has the following options.

Terminal Type

Select the terminal type from the list box. You cannot change the terminal type while the session is connected.

3278	Mainframe (VM, VSE, MVS or z/OS)
3279	Mainframe (VM, VSE, MVS or z/OS)
3270 Printer	3287 printer session
5250	iSeries
5250 Printer	iSeries printer session
VT100	UNIX
VT220	UNIX
ANSI	UNIX

Screen Size

Click the radio button for the desired screen size. For 3279, VT100 or VT220 terminals you can select the **Custom** radio button and define your own screen size from 20 x 80 to 72 x 200. You cannot change the screen size while the session is connected.

Telnet Terminal Type

Only change this field if the host does not accept the generated terminal type. This should rarely need to be done. You may add a single terminal type or a list of terminal types separated by a semi-colon.
For example:

DEC-VT220;VT220;DEC-VT100;VT100

For a list of terminal types, if the first is rejected then the second is sent, and so on until all types have been sent. The example above allows a VT100 connection if VT220 is not supported by the host.

Resource Name

(Optional) (327x terminals) Specify a resource name for your terminal emulation session. You may specify a Windows environment variable. For example, if you specify "%username%" it will be replaced by the Windows username when it is sent to the host. You may also specify a list of resource names. If the first resource is in use, then the session attempts to use the second, and so on. Resource names must be separated by a semicolon. For example,

DEVICE1;DEVICE2;DEVICE3

Device name

(Optional) (5250 terminals) Specify a device name for your terminal emulation session. You may specify a Windows environment variable. For example, if you specify "%username%" it will be replaced by the Windows username when it is sent to the host. You may also specify a list of device names. If the first device name is in use, then the session attempts to use the second, and so on. Device names must be separated by a semicolon. For example,

DEVICE1;DEVICE2;DEVICE3

DEC Answerback

(Optional) (VTxxx terminals) Specify an answerback message. If the host sends a "Transmit answerback message" control character, the answerback message is transmitted back to the host.

Use Computer Name

Check this box to place the network computer name in the resource name edit box.

TN3270E Associate

(Optional) (3270 Printer) Check TN3270E Associate if the resource name species a terminal and you wish to use the printer associated with that

terminal.

3270 Options

Disable TN3270E (327x terminals and 3270 Print) Check this box to negotiate a TN3270 connection instead of a TN3270E connection.

Null/Space Processing (327x terminals) Check this box to activate null/space processing.

Suppress Autowrap (327x terminals) Check this box to suppress autowrap. Autowrap moves the cursor from the last field of the screen to the first field of the screen when moving the cursor forward, or from the first field on the screen to the last field on the screen when moving the cursor backward. This option causes an operator error instead of an autowrap if the user presses the tab key with the cursor in the last field of the display, or the backtab key with the cursor in the first field of the display. It also gives an error if typing fills the last field of the display and autoskip is set.

Whole Field Delete (327x terminals) Check this box to make the Delete key affect the whole field if it wraps from one row to another, instead of just deleting to the end of the row containing the cursor.

Identify input fields (327x terminals) Check this box to display 5250-style column separators in 3270 unprotected fields. This makes it easier to identify the fields that accept input data.

Reset bit unlocks kybd. (327x terminals) Unlock the keyboard when the reset bit (bit 1) in the 3270 Write Control Character (WCC) is on. This option only applies to TN3270E connections.This option is rarely used.

5250 Options

No Column Separators (5250 terminals) Check this box to remove column separators from the display.

Monochrome Display (5250 terminals) Check this box to indicate you want a monochrome display.

Disable Non-blink Cursor (5250 terminals) Check this box to prevent the host from changing the cursor from blink to non-blink.

Disable Auto-help (5250 terminals) Check this box to disable auto-help. Auto-help displays on error message on line 24 when the operator makes a typing error (for example, trying to type into a protected field). When this box is checked, a flashing error code is displayed on line 24 rather than a descriptive error message.

Enhanced Mode (5250 terminals) Check this box to enable support for cursor progression and masked fields.

5250 Print Options

Message Queue Name (Optional) (5250 Print) Specify the AS/400 message queue name.

Message Queue Library (Optional) (5250 Print) Specify the AS/400 message queue library.

Host Print Transform (Optional) (5250 Print) Check this box to enable Host Print Transform for this printer session. When Host Print Transform is enabled, the AS/400 operating system sends printer data formatted specifically for the printer type defined for this printer queue on the AS/400. When Host Print Transform is disabled the AS/400 operating system sends generalized printer data suitable for most printers.

ANSI/VT Options

Prefix LF with CR Check this box to replace a LF (line feed) with a CR/LF (carriage return/line feed combination). Use this option with host computers that send a line feed (LF) instead of a carriage return/line feed (CR/LF) pair.

Strip high-order bit	Converts 8-bit data received from the host computer to 7-bit data.

ANSI Options

Send CR for CR/LF	Check this box to have the CR/LF (carriage return/line feed combination) replaced by a CR (carriage return).
Enable line wrap	Check this box to have lines that extend beyond the last terminal column wrap around to the next line. If this item is not checked the lines are truncated.
Local Echo	Check this box to have typed characters displayed on the screen by TN3270 Plus. Normally characters are echoed back to your emulated terminal by the remote computer, however, if the remote computer does not echo the characters to the screen you can check this option and TN3270 Plus will display typed characters for you.
Scrollback lines	Specify the number of lines to be kept in a buffer to allow scrolling backward.

Null/Space Processing

Overview

Both nulls (X'00') and space characters (X'40') appear as blanks on a display screen. Space characters are transmitted to the host, but nulls are not transmitted as part of the 3270 data stream when modified screen data is sent to the host. Because nulls are not sent to the host, graphic characters may shift positions on the display when the screen is updated by the host.

With null/space processing, nulls are converted to space characters before the data is transmitted. When doing null-to-space conversion, the space will inherit the nulls character attribute. Thus, the converted data appears at the host the same as it does on the display.

Null/Space Processing on Formatted Screens

Null-to-space conversions are performed only on fields modified by the user. Positioning the cursor within a field does not cause conversions to be performed unless the field is modified by the user. In order to reduce transmission time to the host, nulls that follow the last graphic character in a field are not converted to space characters.

Null/Space Processing on Unformatted Screens

Null-to-space conversions are performed only if the data on the screen is modified by the user. In order to reduce transmission time to the host, nulls that follow the last graphic character on the screen are not converted to space characters.

Using Insert Mode with Null/Space Processing

Null/space processing is especially useful in Insert mode. Without null/space processing, when a character string is followed by blanks to the end of a field, characters could be inserted only if those blanks were actually nulls. Spaces at the end of a field had to be changed to nulls before Insert could be used. With null/space processing, those blanks at the end of a field can be either nulls or spaces. Insert mode works with either.

When null/space processing is active and a character is inserted on an unformatted screen, only the characters between the cursor and the end of the screen are shifted to the right. Characters cannot be wrapped past the end of the screen because null-to-space conversions have already been performed on all nulls before the cursor. In fact, with null/space processing active, all leading nulls and embedded nulls within the text are actually converted to spaces before Insert mode is activated.

5.4.5 Colors command

Use this command to change the session colors for the current terminal emulation session. This command displays the Colors pane of the **Session Setup** dialog box. The Colors pane has the following options.

Color Scheme	Select a color scheme from the drop-down list. The list only displays the color schemes for the terminal type specified on the Terminal Pane.
Save As... button	Save the current color scheme with a new name. Color scheme names are **not** case sensitive.
Delete button	Delete the current color scheme.
Field Attribute	Select the **Field Attribute** you wish to change.
Foreground (Text) Color	Click on the desired color. The active field attribute list item foreground color changes to the the new color.
Background Color	Click on the desired color. The active field attribute list item background color changes to the the new color.
More Colors... button	Click the **More Colors...** button if the color you want is not in the color selection panel.
Reset Color button	Resets the selected **Field Attribute** back to its default color.
Reset All button	Resets all field attributes back to their default colors.

5.4.6 Display command

Use this command to change the screen font, cursor style and window options. This command displays the Display pane of the **Session Setup** dialog box. The display pane has the following options.

Screen Font	The current screen font is shown in the edit box. Click the adjacent **Change...** button to display the Font dialog box. and make a new font selection.
Scale font to fit terminal window	Check this box to have the font size adjust to the window size when you resize the window. If this box is not checked, the font size remains the same when you resize the window. Note: Some fonts are not resizable (for example, fixedsys). This option has no affect if the font cannot be resized.
Normal Cursor	Check one of the radio buttons to select the cursor style. Check the Blink check box if you want your cursor to blink. Check the Cross Hair check box of you want a Cross Hair (vertical and/or horizontal rule) added to the cursor.
Insert Mode Cursor	Check one of the radio buttons to select the cursor style to use when you switch to insert mode. Check the Blink check box if you want your cursor to blink. Check the Cross Hair check box of you want a Cross Hair (vertical and/or horizontal rule) added to the cursor.
Cross Hair Style	Check one of the radio buttons to define the appearance of the cross hair cursor. You may specify horizontal rule only, vertical rule only or both horizontal and vertical rules.
Solid terminal window frame	Check this box to have a solid background (instead of the splash screen background) surround the terminal emulation area
Display attribute characters	Check this box to display an at sign (@) in each screen position that contains an attribute byte. Attribute byte tips support is also activated. When this option is on, hovering the cursor over an attribute byte displays a small window containing a description of the attribute byte.
Display blink fields as reverse video	Check this box to display fields with the blink attribute in reverse video instead of blinking.
Display operator information area in terminal window	Check this box to have the operator information

area appear as an additional line in the terminal emulation area.

Disable optimized screen drawing Check this box to disable optimized screen drawing. When this option is disabled, the screen is drawn a single character at a time rather than groups of characters at a time. This option may be required for proper rendering of some bi-directional (BIDI) language fonts. (rarely used)

5.4.7 Printer command

Use this command to change the printer font and print options. This command displays the Printer pane of the **Session Setup** dialog box . The Printer pane contains the following options.

Printer Select the printer to use for this session. Use the drop-down list box to select a printer. Use the **Page Setup...** button to display the Page Layout dialog box and change the printer setup for this session.

The **Application Default** printer selection indicates that the session should use the printer defined in application printer setup (Host, Print Setup...).

<<Disk File: append>> indicates print output should be appended to the file specified in the Filename edit box rather than printed. If the file does not exist, it is created.

<<Disk File: incremental>>indicates print output should be sent to a file rather than printed. A new file will be created for each new print job. The file name is created by incrementing a count and inserting it between the file name and extension specified in the File name edit box. (For example, if you specify a file name of test.txt it will generate files called test.000.txt, test.001.txt, etc.)

Paper Tray (Optional) Select the printer paper tray to use for this session. This option is not available when you specify **<<Disk File: append>>** or **<<Disk File: incremental>>** in the Printer drop-down list box.

Filename (Optional) You must specify **<<Disk File: append>>** or **<<Disk File: incremental>>** in the Printer drop-down list box to enable this edit box. Specify the full path and file name to redirect printed output to a file. Click the **Browse...** button to display the **Select Printer Screen File Name** dialog and choose a file.You may include one or more of the following variables anywhere in the file name and the variable will be replaced with the associated data.

Variable	Case	Substitution
$COMPUTERNAME	(upper case)	Computer name.
$DATE	(upper case)	Current date in YYYY-MM-DD format.
$SESSION	(upper case)	Current session name
$USERNAME	(upper case)	Current logged on user name.

For example, a printer file name of "$DATE Printer.txt" creates a new printer file each day named "YYYY-MM-DD Printer.txt". You may also specify Windows environmental variables (e.g. %computername%, %username%, %appdata%, %programfiles%, etc).

Set Font... Click the adjacent **Set Font...** button to display the Font dialog box. and make a new font selection.

Print Operator Information Area Check this box to include the operator information area (OIA) in the print screen output.

Center Print Screen output on page Check this box to center print screen output on the page rather than starting to print from the upper left-hand corner of the page.

Scale font width to fit page Check this box to scale the font horizontally so that is fits on the page.

Black and White Print Screen Check this box to print the screen in black and white instead of color or grayscale.

Print Screens to Spool Before Printing Check this box to allow multiple Print Screens to be printed on a single printed page. In the edit box following this option, specify the number of Print Screens you want spooled before printing occurs. Print Screen output is spooled to a file until the specified number of print screens is reached, then output is sent to the printer. For example, if you specify 2 Print Screens to be spooled before printing, the first time you issue the print screen command the printed output is spooled. The second time you issue the print screen command, both print screens will be printed. If the spooled Print Screens exceed the size of one printed page, TN3270 Plus will print multiple pages. When this option is active, the status bar displays a printer icon followed by the number of spooled pages.

Display Abort dialog while printing Check this box to display an abort dialog box that allows you to cancel a print screen before it has finished printing.

Send raw data to printer Check this box to send data directly to the printer without formatting. Use this setting if the data contains embedded printer control commands. This is useful for VT100 or VT220 passthrough printing, 5250 printer sessions or 3270 printer sessions.

Append form feed to end of job Check this box if the "Send raw data to printer" option is used and the raw data does not contain a trailing form feed.

Suppress initial form feed Check this box to suppress any form feeds (blank pages) at the beginning of a print job. (3270 printer and 5250 printer sessions only).

Discard SCS transparency blocks Check this box to suppress any SCS (SNA Character String) transparency blocks. This will discard any printer control characters, enabling a report to be printed without the raw data option, or saved as a file without the printer control characters. (3270 printer and 5250 printer sessions only).

Translate EBCDIC in SCS transparency blocks Check this box to perform EBCDIC to ASCII translation in SCS (SNA Character String) transparency blocks. This option is on by default. Uncheck this option to disable EBCDIC to ASCII translation in SCS transparency blocks. (3270 printer and 5250 printer sessions only).

Print null lines as blank lines Check this box to replace null lines with blank lines in printed output. (3270 LU3 printer sessions only).

Enable hex passthrough: Check this box to enable hexadecimal passthrough. The associated **Escape string** edit box will accept one or two strings of any length, separated by a comma. The first string initiates passthrough processing and the second sting terminates passthrough processing (for example , **%%,&&**). If you just enter one string then it's used as both the initializer and the terminator (for example, **%%**). Hexadecimal passthrough allows you pass hexadecimal data to the printer. This data is typically printer control sequences that allow control printer operation. You can skip to top of page or change fonts etc. The data in the data stream between the initiator and terminator is treated as hexadecimal data and passed directly to the printer. For example, if you wanted to send 0x1B to the printer and your initiator was **%%** and your terminator was **&&** you would place **%%1B&&** in your data stream. (3270 printer and 5250 printer sessions only).

Print Screen Header (Optional) Specify a header for the print screen output. The header and footer accept the following variables:

Variable	Case	Substitution
$COMPUTERNAME	(upper case)	Computer name.
$DATE	(upper case)	Current locale date (mm/dd/yy)
$LONGDATE	(upper case)	Current locale long date (day month dd, yyyy)
$SESSION	(upper case)	Current session name
$TIME	(upper case)	Current locale time (hh/mm/ss)
$USERNAME	(upper case)	Current logged on user name.

For example - a header of "Sample output $DATE $TIME" would expand to: "Sample output 05/21/00 20:40:52." You may also specify Windows environmental variables (e.g. %computername%, %username%, %appdata%, %programfiles%, etc). (Terminal sessions only.)

Print Screen Footer (Optional) Specify a footer for the print screen output. The footer accepts the same variables as the header. (Terminal sessions only.)

Printer init. string (Optional) Specify a printer initialization string. Use this field to specify hexadecimal printer control codes to initialize the printer. You may include spaces in this field to improve readability. This string is sent to the printer before each print job. This field is only available when the "Send raw data to printer" option is checked. (3270 printer and 5250 printer sessions only).

Printer term. string (Optional) Specify a printer termination string. Use this field to specify hexadecimal printer control codes to terminate the printer. You may include spaces in this field to improve readability. This string is sent to the printer after each print job. This field is only available when the "Send raw data to printer" option is checked. (3270 printer and 5250 printer sessions only).

5.4.8 Keyboard command

Use this command to change the keyboard mapping for the current terminal emulation session. This command displays the Keyboard pane of the **Session Setup** dialog box. The Keyboard pane contains the following options.

Keyboard Map Name A default keyboard map is provided for each terminal type supported by TN3270 Plus. You may modify the default keyboard map or build your own customized keyboard map(s) by changing the default keyboard map and saving it with a new name. To modify an existing keyboard map or to create a new keyboard map, click the **Configure...** button to display the Keyboard Map Setup dialog box. Once you have created a customized keyboard map, you can select it from the **Keyboard Map Name** drop-down list box.

Type Ahead The **Buffer Size** edit box contains the size of the type-ahead buffer. You may increase or decrease the size of the type-ahead buffer by changing this value. Check the **Disable** check box to disable the type-ahead buffer support.

Destructive Backspace Check this box to enable destructive backspace. With destructive backspace enabled, the backspace key will delete a character as it moves the cursor over it. With destructive backspace disabled, the backspace key will not delete a character; it will simply move the cursor.

Respect numeric fields Check this box to respect numeric fields. This means that only numeric characters can be typed into 3270 screen fields with the numeric field attribute turned on. When this option is on, you can override the numeric field requirement by holding down the shift key and entering non-numeric characters. (3270 sessions only).

Lock keyboard on operator error Check this box if you want the keyboard locked after a data entry error. Remove the check to prevent locking of the keyboard. The default is unchecked (3270 and 5250 sessions only).

Suppress alarm on operator error Check this box to suppress the beep if an operator types into a protected field or presses an invalid keyboard combination. The default is unchecked (3270 and 5250 sessions only).

Typematic AID keys Check this box to make all AID generating keys typematic (repeating). However, the key will only be repeated when the host unlocks the keyboard. This is to prevent the type-ahead buffer from filling up with AID keys while the host is processing the last key (3270 and 5250 sessions only).

Don't reset insert mode on Enter/AID key Check this box to prevent insert mode from being reset (turned off) after the Enter key or an AID key is pressed (3270 and 5250

sessions only).

Upper-case input Check this box to make all keyboard input appear in upper case regardless of the CAPS lock setting.

Disable application cursor keys mode Check this box to disable sending alternate escape sequences for the normal arrow keys. This option is rarely used. (VT100, VT220 and ANSI sessions only).

Disable application keypad mode Check this box to to disable sending alternate escape sequences for the keypad keys. For example, use this option if the keypad keys do not work correctly with the vi editor. This option is rarely used. (VT100, VT220 and ANSI sessions only).

5.4.9 Toolbar command

Use this command to change the toolbar configuration. This command displays the Toolbars pane of the Session Setup dialog box. The Toolbar pane contains the following options.

Standard toolbar Check this box to display the standard toolbar.

Status bar Check this box to display the status bar.

Keypad Toolbar Check this box to display the keypad toolbar.

Keypad Toolbar Name A default keypad toolbar is provided for each terminal type supported by TN3270 Plus. You may modify the keypad toolbar or build your own customized keypad toolbar by changing the default keypad toolbar and saving it with a new name. To modify an existing keypad toolbar or to create a new keypad toolbar, click the **Customize...** button to display the Customize Toolbar dialog box. Once you have created a customized keypad toolbar, you can select it from the Keypad **Toolbar Name** drop-down list box.

Fixed width; number of columns Specify the number of columns for the keypad toolbar. Specify 0 to make the toolbar dynamically resizable (by dragging its frame).

Dockable Check this box to allow the keypad toolbar to be docked with the TN3270 Plus window.

Custom Toolbar Check this box to display the custom toolbar.

Custom Toolbar Name A default custom toolbar is provided for each terminal type supported by TN3270 Plus. Use the custom toolbar to create button shortcut to commonly used scripts, macros or commands. You may modify the default custom toolbar or build your own customized custom toolbar by changing the default custom toolbar and saving it with a new name. To modify an existing custom toolbar or to create a new keypad toolbar, click the **Customize...** button to display the Customize Toolbar dialog box. Once you have created a customized keypad toolbar, you can select it from the Custom **Toolbar Name** drop-down list box.

Fixed width; number of columns Specify the number of columns for the custom toolbar. Specify 0 to make the toolbar dynamically resizable (by dragging its frame).

Dockable Check this box to allow the custom toolbar to be docked with the TN3270 Plus window.

5.4.10 Proxy Server command

Use this command to configure TN3270 Plus to connect to a host computer via a proxy server or firewall. This command displays the Proxy Server Pane of the **Session Setup** dialog box. the Proxy Server pane contains the following options.

None Check this box if you are not connecting via a proxy server.

SOCKS 4 Protocol Check this box if your proxy server uses the SOCKS version 4 proxy server protocol.

SOCKS 4A (proxy can resolve host names) Check this box if your proxy server uses the SOCKS version 4A proxy server protocol.

Telnet Check this box if your proxy server uses the Telnet proxy server protocol.

HTTP Check this box if your proxy server uses the HTTP proxy server protocol. After connecting to the proxy server the following data is sent:
CONNECT *host*::*port* HTTP/1.1
Host *host*::*port*
Proxy-Authorization: Basic *username*:*password*

Proxy Name/IP Enter the domain name or IP address of the proxy server.

Proxy Port Enter the proxy server port number.

Username (Optional) Enter the user name required by the proxy server. This field is only valid for the HTTP proxy server protocol.

Password (Optional) Enter the password required by the proxy server. This field is only valid for the HTTP proxy server protocol.

Connect Command (Optional) This field is used differently for non-SOCKS compliant and SOCKS compliant proxy servers.

For non-SOCKS compliant proxy servers, enter the connection command required by the proxy server. You can use the following variables in your connect command:

$HOST	Replaced with the Host Name from the Host Pane.
$PORT	Replaced with the Port Number from the Host Pane.
<CR>	Sends a carriage return to the proxy server.
<LF>	Sends a line feed to the proxy server.

For example, WinGate, a popular proxy server/firewall, prompts the user for the host name and port number. The following "Connect Command" string works with WinGate:

$HOST:$PORT<CR><LF>

For a SOCKS compliant proxy server, enter the USERID for SOCKS authentication.

WinGate™ is a registered trademark of Deerfield.com (http://wingate.deerfield.com/)

5.4.11 Security command

Use this command to configure TN3270 Plus to create a secure connection to a host computer. This command displays the Security pane of the **Session Setup** dialog box. The command requires the SSH and/or SSL feature. This Security pane contains the following options.

Encryption Protocol: Select the desired encryption protocol in the drop-down list box. "None" is the default. After the protocol is selected, configuration options are displayed.

Protocol	Description
None	No encryption. This is the default setting.
SSLv2	Secure Socket Layer version 2. (Warning: We do not recommend the use of the SSL version 2 protocol because it contains some known security issues.)

SSLv3	Secure Socket Layer version 3. (Warning: We do not recommend the use of the SSL version 3 protocol because it contains some known security issues.)
TLSv1	Transport Layer Security version 1.0
TLSv1.1	Transport Layer Security version 1.1
TLSv1.2	Transport Layer Security version 1.2
SSHv1	Secure Shell version 1. (Warning: We do not recommend the use of the SSH version 1 protocol because it contains some known security issues.)
SSHv2	Secure Shell version 2.
SSHv1 or SSH v2	TN3270 Plus negotiates either SSHv1 or SSHv2 with the host computer.

Ciphers Button: Click the **Ciphers** button to select the ciphers that may be used for the SSL or TLS handshake. The **Ciphers** button displays the **Specify Ciphers** dialog box. In the **Specify Ciphers** dialog box, use the **Add ->** and **<-Remove** buttons to add ciphers to, or remove ciphers from, the "Selected Ciphers" list box. Only ciphers in the "Selected Ciphers" list box are available for selection in the SSL or TLS handshake. Use the **Remove all CBC ciphers** button to remove all cipher-block chaining (CBC) mode ciphers from the list of selected ciphers. Removing the CBC mode ciphers eliminates the POODLE vulnerability from SSLv3 connections.

SSL or TLS Configuration Options

Host will initiate SSL connection (Optional) If this option is checked, TN3270 Plus waits for the host to send the STARTTLS option instead of initiating the SSL handshake immediately after connection.

Display certificate when connected (Optional) If this option is checked, TN3270 Plus displays the Server Certificate Details dialog and allows you to accept or reject the certificate.

Accept self-signed certificates (Optional) If this option is checked, TN3270 Plus accepts a self-signed server certificate. If this option is not checked, a dialog box prompts you to accept or reject a self-signed certificate. If you accept the self-signed certificate, the connection is completed and "Accept self-signed certificates" is set on for the session, so the self-signed certificate is accepted without prompting the next time you connect. This option applies to server certificates not client certificates.

Accept expired certificates (Optional) If this option is checked, TN3270 Plus accepts expired SSL server certificates. If this option is not checked, a dialog box prompts you to accept or reject an expired certificate. If you accept the expired certificate, the connection is completed and "Accept expired certificates" is set on for the session, so the expired certificate is accepted without prompting the next time you connect. This option applies to server certificates not client certificates.

Accept certificates not yet valid (Optional) If this option is checked, TN3270 Plus accepts server certificates that have not reached their effective date. If this option is not checked, a dialog box prompts you to accept or reject the not yet valid certificate. If you accept the not yet valid certificate, the connection is completed and "Accept certificates not yet valid" is set on for the session, so the not yet valid certificate is accepted without prompting the next time you connect. This option applies to server certificates not client certificates.

Accepts ANY invalid certificate (Optional) If this option is checked, TN3270 Plus accepts any invalid server certificates. This option applies to server certificates not client certificates.

Server Certificate File (.pem) (Optional) Enter the server certificate file name (.pem). Click the **Browse...** button to display the **Select Certificate Filename** dialog box. In order for TN3270 Plus to validate the server certificate, this file must contain the entire server certificate chain in the proper order starting with the server certificate and ending with the root Certificate Authority certificate. You can use notepad merge certificates into a single file.

Use Windows certificate store (Optional) If this option is checked, TN3270 Plus imports all the certificates from the Internet Explorer Trusted Root Certification Authorities certificate store into the TN3270 Plus certificate store (tn3270.pem). These certificates are then used to validate the SSL certificate chain. If you add certificates to the Internet Explorer Trusted Root Certification Authorities certificate store click the **Refresh** button and TN3270 Plus add the new certificates to its certificate store.

Client Certificate File (.pem) (Optional) Enter a client certificate file name (.pem). Click the **Browse...** button to display the **Select Client Certificate Filename** dialog box.

Password (Optional) Specify a client certificate encryption password.

SSH Configuration Options

Authentication Type: Choose the SSH authentication type.

Password

Keyboard-interactive

Public Key

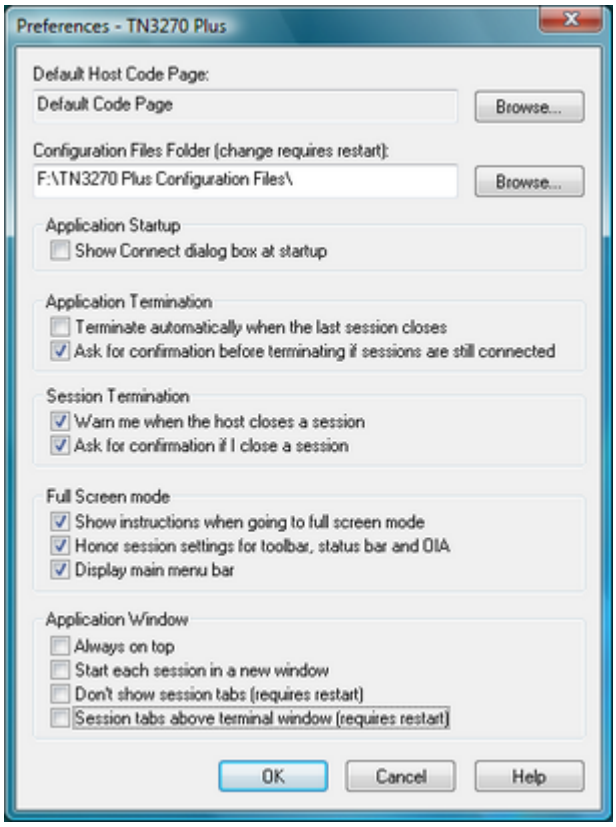
Username: (Optional) Specify your username. TN3270 Plus will remember the username. You will be prompted for your password. If you leave this field blank, you will be prompted for both your username and password.

Key regeneration interval (minutes): (Optional) Specify a key regeneration interval. This specifies how many minutes you want to elapse before key regeneration is initiated. The default is 60 minutes. If an SSH key generated at connection time is used for too long, it is possible that the key could be compromised. The SSHv2 protocol specifies that a new key should be generated periodically. This regeneration may be initiated by the client or the host. TN3270 Plus initiates key regeneration each time the key regeneration interval expires. Data transfer stops while the key is regenerated and renegotiated with the host so there is a performance penalty for doing key regeneration too often. The default key regeneration interval of 60 minutes is the recommended interval.

Private key file for authentication: (Optional) Specify a private key file when using public key authentication. This key may be in Putty format (*.ppk), OpenSSH format or IETF format.

5.4.12 Preferences command

Use this command to personalize TN3270 Plus. This command displays the **Preferences** dialog box:



The Preferences dialog box has the following options:

Default Host Code Page

The edit box indicates the default language code page. The default language code page is used for sessions that have not explicitly specified a code page file (Setup, Sessions..., Host, Code Page). Click the Browse... button to display the Code Page Setup dialog box. To specify a new default language code page from the Code Page Setup dialog box,.select a code page from the drop-down list box or click the Browse button to choose an *.cpg file.

Configuration File Folder (change requires restart)

The edit box contains the folder location of the TN3270 Plus configuration files. See Moving the TN3270 Plus Configuration Files for instructions about moving the configuration files to a new folder.

Application Startup

- Show Connect dialog box at startup

  Check this box to display the **Connect to Host** dialog each time TN3270 Plus is started. If TN3270 Plus is configured to start sessions automatically, you may choose not to display the **Connect to Host** dialog when TN3270 Plus is started.

Application Termination

- Terminate automatically when the last session closes

  Check this box to terminate TN3270 Plus when the last session is closed. If this box is not checked, TN3270 Plus will remain active and display the splash screen after the last session is closed.

- Ask for confirmation before terminating if sessions are still connected

Check this box and TN3270 Plus asks if you really want to exit while sessions are still active (Host, Exit). If this box is not checked,
TN3270 Plus will terminate while sessions are active.

Session Termination

- Warn me when the host closes a session

  Check this box to display an information dialog box when the host computer closes a session.

- Ask for confirmation if I close a session

  Check this box to display a confirmation dialog box when you try to close a session (Host, Close).

Full Screen mode

- Show instructions when going to full screen mode

  Check this box to display the Full Screen Mode confirmation dialog box when you switch to full screen mode.

- Honor session settings for toolbar, status bar and OIA

  By default, Full Screen Mode removes the window frame, toolbar and status bar and adds the Operator Information Area (OIA). Check this box to use your session settings rather than the default Full Screen Mode settings.

- Display main menu bar

  By default, Full Screen Mode removes the menu bar from the display. Check this box to display the menu bar in Full Screen Mode.

Application Window

- Always on top

  Check this box to keep the application window visible even when another application has the focus.

  Warning: If this option is set when in Full Screen Mode then no other application windows are accessible until full screen mode is ended.

- Start each session in a new window

  Check this item to start new sessions in their own windows. When this option is checked, a new window is created for each session. When this option is checked, it takes precedence over the session option (Setup, Sessions…, Setup Items = Host, Start in a new window). If this option is not checked, the session option determines whether or not a new window is created.

- Don't show session tabs (requires restart)

  Check this item to eliminate session tabs from the TN3270 Plus window. When this option is checked session tabs never appear. When this option is not checked, session tabs appear when you connect more than one session in a window. TN3270 Plus must be restarted for this option to take effect. See Working with Multiple Sessions for more information about connecting multiple sessions.

- Session tabs above terminal window (requires restart)

  Check this item to move the session tabs above the TN3270 Plus terminal window. When this option is not checked, session tabs appear below the TN3270 Plus terminal window. TN3270 Plus must be restarted for this option to take effect. See Working with Multiple Sessions for more information about connecting multiple sessions.

5.4.13 Trace Socket command

Use this diagnostic command to trace socket activity. Normally you will not use this command unless requested to do so by SDI technical support.

To run a trace manually, perform the following steps:

1. Start TN3270 Plus.
2. Start the trace.
   a) Open the TN3270 Plus **Setup** menu and click on **Trace Socket...**
   b) In the **Select Trace File Name** dialog box, specify a name for the trace file and note the folder that it is in.
3. Connect to the host computer and run the failing application.
4. Stop the trace. Open the TN3270 Plus **Setup** menu and click on **Trace Socket...**
5. E-mail the trace file and a description of the problem to support@sdisw.com.

To automatically activate a trace when TN3270 Plus is started :

Enter the following registry entries.

DWORD: HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\TraceAtStartup = 1
STRING: HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\TraceFilePath

Set TraceFilePath to the full path and filename of the trace output file.

5.5 Macros Menu

5.5.1 Command Summary

The Macros menu contains the following commands:

Start Recording	Start the macro recorder.
Stop Recording	Stop the macro recorder.
Replay	Select and play a macro.
Edit	Select and edit a macro.
Delete	Delete a macro.
Assign Key	Assign a shortcut key to a macro.
Cancel Replay	Cancel the macro that is playing.

5.5.2 Start Recording command

Use this command to start the macro recorder. The macro recorder records your keystrokes so they can be replayed later.

For a tutorial on the macro recorder, see Using the Macro Recorder.

System Command:

MacroStartRecording

5.5.3 Stop Recording command

Use this command to stop the macro recorder. After clicking Stop Recording, the Save As dialog box prompts you for a macro name. Enter a macro name and click the OK button to save the macro.

For a tutorial on the macro recorder, see Using the Macro Recorder.

System Command:

MacroStopRecording

5.5.4 Replay command

Use this command to replay a macro. Move your mouse over the Replay menu item and a second level menu of macros is displayed. Click on the macro you want to play.

For a tutorial on the macro recorder, see Using the Macro Recorder.

5.5.5 Edit command

Use this command to edit a macro. Move your mouse over the Edit menu item and a second level menu of macros is displayed. Click on the macro you want to edit. The Macro Editor dialog box is displayed.

For a tutorial on the macro recorder, see Using the Macro Recorder.

5.5.6 Delete command

Use this command to delete a macro. Move your mouse over the Delete menu item and a second level menu of macros is displayed. Click on the macro you want to delete.

For a tutorial on the macro recorder, see Using the Macro Recorder.

5.5.7 Assign Key command

Use this command to assign a keyboard shortcut to a macro.

Use the following steps to assign a shortcut key:

1. Move your mouse over the Assign Key menu item and click on a macro in the second level menu.
2. The Type Key dialog box is displayed.
3. Type the desired key combination or position the mouse over the instructional text and click a mouse button. (You can use a combination that includes a keyboard key and a mouse click.)
4. Click the OK button to complete the key assignment.
5. Click the Retry button to clear any current setting so that you can select a different key combination.
6. Click the Cancel button to exit the Type Key dialog box without making an assignment.

For a tutorial on the macro recorder, see Using the Macro Recorder.

5.5.8 Cancel Replay command

Use this command to cancel a macro while it is playing. This command cancels and active macro in the currently active session.

5.6 Language Menu

5.6.1 Language Items

Select the desired language. Changing the language changes the language in the toolbars, menus and dialog boxes.

5.7 Help Menu

5.7.1 Command Summary

The Help menu contains the following commands, which provide you assistance with this application:

Help Topics	Displays the table of contents for the help file. This command also gives you access to the help index and the help find function.
About TN3270 Plus...	Displays the version number and build date of this application.
Registration Form	Displays the TN3270 Plus registration form.
Purchase Online	Opens your default web browser to the TN3270 Plus Secure Order System.

5.7.2 Help Topics

Use this command to display the contents of the TN3270 Plus Help file. From the opening screen, you can jump to step-by-step instructions for using TN3270 Plus and various types of reference information.

Once you open Help, you can click the Contents button whenever you want to return to the opening screen.

5.7.3 About TN3270 Plus

Use this command to display the copyright notice, version number and build date of your copy of TN3270 Plus.

5.7.4 Purchase Online

Use this command to purchase TN3270 Plus with a credit card (Visa, MasterCard/Eurocard, American Express, Discover/Novus, Diners Club/Carte Blanche and JCB) or PayPal via the Internet. This command opens your default web browser to the TN3270 Plus Secure Order System, http://sdisw.com/tn3270RegInst.html.

5.7.5 Check for Latest Version

Use this command to check if you are running the most recent version of TN3270 Plus. If you are not running the current version, you will be asked if you want to download the current version.

6 Reference

6.1 Default Keyboard Maps

6.1.1 Default 3270 Keyboard Map

Attn	Alt+A
Backspace	Bksp
Backtab	Shift+Tab
CentSign	Shift+Alt+4
Clear	Pause, Esc, Alt+C

CursorDown	Down	move cursor down
CursorLeft	Left	move cursor left
CursorMove	MouseLeftClick	move the cursor to the mouse location
CursorRight	Right	move cursor right
CurorSelect	MouseLeftDoubleClick	for light pen detectable fields

CursorUp	Up	up arrow
Delete	Del, NumDel	
Dup	Alt+D	
EditSelectionDown	Shift+Down	select for cut or copy
EditSelectionLeft	Shift+Left	select for cut or copy
EditSelectionRight	Shift+Right	select for cut or copy
EditSelectionUp	Shift+Up	select for cut or copy
EditSelectWord		select a word for cut or copy
Enter	Enter, NumEnter, RightCtrl	
EraseEof	Shift+End, Shift+NumEnd	
EraseInput	Shift+Del, Shift+NumDel	
FastCursorDown	NumDown	move cursor rapidly down
FastCursorLeft	NumLeft	move cursor rapidly left
FastCursorRight	NumRight	move cursor rapidly right
FastCursorUp	NumUp	move cursor rapidly up
FieldEnd	End, NumEnd	
FieldMark	Alt+F	
Home	Home, NumHome	
Hotspot		activate Hotspot support
Insert	Ins	
InsertOn	NumIns	
Newline	Shift+Enter, Shift+NumEnter	
PA1	Alt+F1, PageUp, NumPageUp	
PA2	Alt+F2, PageDown, NumPageDown	
PA3	Alt+F3	
PF1 - PF6	F1 - F6	
PF7	F7, Shift+PageUp, Shift+NumPageUp	
PF8	F8, Shift+PageDown, Shift+NumPageDown	
PF9 - PF12	F9 - F12	
PF13 - PF24	Shift+F1 - Shift+F12	
Reset	Left Ctrl, Alt+R	
SystemRequest	Alt+S	
Tab	Tab	
TestRequest	Alt+T	
ToggleCrossHairCursor	Alt+X	

Entry Assist

WordDelete	Alt+Del	Delete a word.
WordLeft	Alt+Left	Move the cursor to the next unprotected word
WordLeftAny		Move the cursor to the next word
WordRight	Alt+Right	Move the cursor to the previous unprotected word
WordRightAny		Move the cursor to the previous word

6.1.2 Default 5250 Keyboard Map

Attn	Alt+A	
Backspace	Bksp	
Backtab	Shift+Tab	
CentSign	Shift+Alt+4	
Clear	Esc, Alt+C	
CursorDown	Down	move cursor down
CursorLeft	Left	move cursor left
CursorMove	MouseLeftClick	move the cursor to the mouse location
CursorRight	Right	move cursor right
CursorSelect	MouseLeftDoubleClick	for light pen detectable fields
CursorUp	Up	move cursor up
Delete	Del, NumDelete	
Dup	Alt+D	
EditSelectionDown	Shift+Down	select for cut or copy
EditSelectionLeft	Shift+Left	select for cut or copy
EditSelectionRight	Shift+Right	select for cut or copy
EditSelectionUp	Shift+Up	select for cut or copy
EditSelectWord		select a word for cut or copy
Enter	Enter, RightCtrl	
EraseEof	Shift+End, Shift+NumEnd	
EraseInput	Shift+Del, Shift+NumDel	
ErrorReset	LeftCtrl	
FastCursorDown	NumDown	move cursor rapidly down
FastCursorLeft	NumLeft	move cursor rapidly left
FastCursorRight	NumRight	move cursor rapidly right
FastCursorUp	NumUp	move cursor rapidly up
FieldEnd	End, NumEnd	
FieldExit	NumEnter	
FieldMark	Alt+F	

FieldMinus	Num-	Keypad minus sign
FieldPlus	NumAdd	
F1 - F12	F1 - F12	
F13 - F24	Shift+F1 - Shift+F12	
Help	Pause	
Home	Home, NumHome	
Hotspot		activate Hotspot support
Insert	Ins, NumIns	
InsertOn		
Newline	Shift+Enter	
PA1		
PA2		
PA3		
Print		
RollDown/PageUp	PageUp, NumPageUp	
RollUp/PageDown	PageDown, NumPageDown	
SystemRequest	Alt+S	
Tab	Tab	
ToggleCrossHairCursor	Alt+X	
Entry Assist		
WordDelete	Alt+Del	Delete a word.
WordLeft	Alt+Left	Move the cursor to the next unprotected word
WordLeftAny		Move the cursor to the next word
WordRight	Alt+Right	Move the cursor to the previous unprotected word
WordRightAny		Move the cursor to the previous word

6.1.3 Default VT100 Keyboard Map

Backspace	Bksp	
Backtab		
Clear		
CursorDown	Down	
CursorDownApplMode	Alt+Down	
CursorLeft	Left	
CursorLeftApplMode	Alt+Left	
CursorMove	MouseLeftClick	move the cursor to the mouse location
CursorRight	Right	

CursorRightApplMode	Alt+Right	
CursorUp	Up	
CursorUpApplMode	Alt+Up	
Delete	Del	
EditSelectionDown	Shift+Down	select for cut or copy
EditSelectionLeft	Shift+Left	select for cut or copy
EditSelectionRight	Shift+Right	select for cut or copy
EditSelectionUp	Shift+Up	select for cut or copy
EditSelectWord		select a word for cut or copy
Enter	Enter	
Escape	Esc	
Hotspot		activate Hotspot support
Keypad0 to Keypad9	Num0 to Num9	
Keypad0ApplMode	NumIns	
Keypad1ApplMode	NumEnd	
Keypad2ApplMode	NumDown	
Keypad3ApplMode	NumPageDown	
Keypad4ApplMode	NumLeft	
Keypad5ApplMode	NumCenter	
Keypad6ApplMode	NumRight	
Keypad7ApplMode	NumHome	
Keypad8pplMode	NumUp	
Keypad9ApplMode	NumPageUp	
KeypadComma	NumAdd	
KeypadCommaApplMode	Shift+NumAdd	
KeypadDecimal	Num.	
KeypadDecimalApplMode	NumDel	
KeypadEnter	NumEnter	
KeypadEnterApplMode	Shift+NumEnter	
KeypadMinus	Num-	
KeypadMinusApplMode	Shift+Num-	
LineFeed	Shift+Enter	
Null	Ctl+Space	
PF1	F1	
PF2	F2	
PF3	F3	
PF4	F4	
ScrollLineUp	Ctl+Up	
ScrollLineDown	Ctl+Down	
ScrollPageUp	Ctl+PageUp	

ScrollPageDown	Ctl+PageDown
ScrollHome	Ctl+Home
ScrollEnd	Ctl+End
Tab	Tab
ToggleCrossHairCursor	Alt+X

F1 - F4	Ctl+F1 - Ctl+F4	
F5 - F12	F5 - F12	
F13 - F20	Shift+F1 - Shift+F8	
F21- F60		
UDK1-UDK20		Keys for user definition.

Home(Find)	Home
End(Select)	End
Insert	Ins
Del(VT220)	Shift+Del
PageUp	PageUp
PageDown	PageDown

TelnetAbortOutput
TelnetAreYouThere
TelnetBreak
TelnetEraseCharacter
TelnetEraseLine
TelnetInterruptProcess

6.2 Dynamic Data Exchange (DDE)

6.2.1 DDE Functions

TN3270 Plus can function as a Dynamic Data Exchange (DDE) server. This allows DDE client applications to communicate with TN3270 Plus. For example, some users write Microsoft Excel macros to capture data from the TN3270 Plus terminal window and populate a spreadsheet. TN3270 Plus supports the following DDE functions:

Session Topic Functions

Use the session topic to send data to, and receive data from, a terminal session.

Connect Service=tn3270,Topic=*session name* | session#*session_number*

Examples:

Connect Service=tn3270,Topic=mainframe
Connect Service=tn3270,Topic=session#3

Return Values Success: A handle to the conversation.
Failure: 0

Execute Item=close closes the session.

	Item=macro *macroname*	executes a macro. ***macroname*** is the macro name.
	Item=script *filename*	executes a script. ***filename*** is the full path to the script to be executed.
	Return Values	Success: (HDDEDATA) DDE_FACK Failure: (HDDEDATA) DDE_FNOTPROCESSED
Request	**Item=PS**	returns the presentation space (screen image).
	Item=Cursor	returns the cursor position (relative to 1).
	Item=Rows	returns the number of rows in the presentation space.
	Item=Columns	returns the number of columns in the presentation space.
	Item=Emulator	returns the window handle of the TN3270 Plus window.
	Item=Keyboard	returns "Locked" if the keyboard is locked or else it returns "Clear".
	Item=OIA	returns the Operator Information Area (note that this is NOT in IBM format).
	Item=R*nn*	returns the nth row in the presentation space.
	Item=F*nn*	returns the nth field in the presentation space.
	Item=F*nn*U	returns the nth unprotected field in the presentation space.
	Item=F*nn*P	returns the nth protected field in the presentation space.
	Item=P*nnn*[F|L*mmm*]	returns a portion of the presentation space. **P** = Identifier for starting position ***nnn*** = starting position on screen (relative to 1) **F** = return entire field at the starting position **L** = Identifier for length ***mmm*** = length of data to return **Examples:** **P560** returns screen data from position 560 to the end of the field. **P560F** returns the entire field encompassing position 560. **P560L80** returns data from position 560 for a length of 80 characters.
	Return Values	Success: A handle to the data being returned. Failure: 0
Poke	**Item=Keystroke**	sends one or more keystrokes to TN3270 Plus (same format as WinHLLAPI). See the Keystroke Table for a list of all the keystrokes.
	Item=Cursor	moves the cursor position (relative to 1).
	Item=EscChar	sets the escape character for Item=Keystroke (default is '@').
	Return Values	Success: (HDDEDATA) DDE_FACK Failure: (HDDEDATA) DDE_FNOTPROCESSED
Advise	**Item=PS**	sends the presentation space to the client whenever it is updated.
	Item=Cursor	sends the cursor position to the client whenever it is updated.
	Return Values	Success: (HDDEDATA) TRUE Failure: (HDDEDATA) FALSE

System Topic Functions

Use the system topic to send instructions to the TN3270 Plus application.

Connect Service=tn3270,Topic=system

	Return Values	Success: A handle to the conversation. Failure: 0
Execute	**Item=connect *session***	connects to a session. ***session*** is the name of the session.
	Item=close *session*	closes a session. ***session*** is the name of the session.
	Item=key *keyname*	sends a menu key. ***keyname*** is the name of the menu key.
	Item=script *filename*	executes a script. ***filename*** is the full path to the script to be executed.
	Return Values	Success: (HDDEDATA) DDE_FACK Failure: (HDDEDATA) DDE_FNOTPROCESSED
Request	**Item=activesession**	returns the session number of the currently active session.
	Item=sessions	returns the total number of active sessions.

Tnrun.exe

The tnrun.exe program can execute DDE functions from an external script or program (e.g. Windows Script). The command line format is:

```
tnrun.exe topic command item
tnrun.exe topic request {/e environment_variable_name | /c} item
```

Parameters that contain spaces must be enclosed in double quotes.

Where:

topic	command	item
SYSTEM	connect | start	session_name
SYSTEM	disconnect | close	session_name
SYSTEM	key	keyname
SYSTEM	script	filename
session_name	disconnect | close	
session_name	key	keyname
session_name	macro	macroname
session_name	poke cursor	cursor_address
session_name	poke EscChar	escape_character
session_name	poke keystroke	keystroke
session_name	script	filename

/e environment_variable_name for DDE requests the returned data is placed in the specified environment

variable.

/c for DDE requests the returned data is placed on the clipboard in text format.

Examples:

`tnrun.exe SYSTEM connect mainframe`	(Start the TN3270 Plus session "mainframe".)
`tnrun.exe mainframe disconnect`	(Disconnect the TN3270 Plus "mainframe" session.)
`tnrun.exe mainframe macro login`	(Run the "login" macro on session mainframe")
`tnrun.exe iSeries keystroke @E`	(Send the enter key to the "iSeries" session.)
`tnrun.exe "PRD A" script "c:\scripts\Log me in"`	(Run the "Log me in" script in session "PRD A".)
`tnrun.exe mainframe request /c PS`	(Copy the mainframe session presentation space to the clipboard.)

Sample Call to Tnrun.exe using VBScript

```
' CallTnrun.vbs
' Language: VBScript using the Windows Script Host
' When the objShell.Run command parameter contains double quotes within the parameter,
' substitute each double quote in the parameter with two double quotes.
dim objShell
set objShell = WScript.CreateObject("WScript.Shell")
objShell.Run """c:\Program Files (x86)\SDI\TN3270 Plus\tnrun.exe"" SYSTEM connect mai
objShell.Run """c:\Program Files (x86)\SDI\TN3270 Plus\tnrun.exe"" mainframe macro log
```

Sample Programs

Sample programs are available for download from the Customer Downloads web page.

6.2.2 DDE Keystroke Table

@	@@	x	x		
Alt	@A				
Attention	@A@Q	x	x		
Backspace	@<	x	x	x	x
Backtab (Left Tab)	@B	x	x	x	x
Clear	@C	x	x	x	x
Cursor Down	@V	x	x	x	x
Cursor Left	@L	x	x	x	x
Cursor Right	@Z	x	x	x	x
Cursor Select	@A@J	x	x		
Cursor Up	@U	x	x	x	x
Delete	@D	x	x	x	
DEL (VT220)	@A@D				x
Dup	@S@x	x	x		
End	@q	x	x	x	x
Enter	@E	x	x	x	x
Erase EOF	@F	x	x		
Erase Input	@A@F	x	x		
Escape	@A@e			x	x
Field Exit	@A@E		x		

Field Mark	@S@y	x	x		
Field -	@A@-		x		
Field +	@A@+		x		
Help	@H		x		
Home	@0 *(zero)*	x	x	x	x
Insert	@I	x	x	x	x
Insert Toggle	@A@I	x	x		
Host Print	@A@t		x		
Left Tab (Backtab)	@B	x	x		
New Line	@N	x	x	x	x
Page Up	@u		x	x	x
Page Down	@v		x	x	x
Record Backspace	@A@<		x		
Reset	@R	x			
Right Tab (Tab)	@T	x	x	x	x
Shift	@S				
Sys Request	@A@H	x	x		
Tab (Right Tab)	@T	x	x	x	x
Test	@A@C	x	x		
PA1	@x	x			
PA2	@y	x			
PA3	@z	x			
PF1 (VT100)	@A@1			x	x
PF2 (VT100)	@A@2			x	x
PF3 (VT100)	@A@3			x	x
PF4 (VT100)	@A@4			x	x
PF1/F1	@1	x	x		x
PF2/F2	@2	x	x		x
PF3/F3	@3	x	x		x
PF4/F4	@4	x	x		x
PF5/F5	@5	x	x		x
PF6/F6	@6	x	x		x
PF7/F7	@7	x	x		x
PF8/F8	@8	x	x		x
PF9/F9	@9	x	x		x
PF10/F10	@a	x	x		x
PF11/F11	@b	x	x		x
PF12/F12	@c	x	x		x
PF13	@d	x	x		x
PF14	@e	x	x		x
PF15	@f	x	x		x
PF16	@g	x	x		x
PF17	@h	x	x		x
PF18	@i	x	x		x

PF19	@j	x	x	x
PF20	@k	x	x	x
PF21	@l	x	x	x
PF22	@m	x	x	x
PF23	@n	x	x	x
PF24	@o	x	x	x

* Note: The DDE keystroke mnemonics are case sensitive!

Sample Programs

Sample programs are available for download from the Customer Downloads page on our web site.

6.2.3 DDE in an Excel Macro

The following is a sample Excel VB macro using DDE to pass data from Excel to TN3270 Plus.

```
Sub DDESample()

    ' Initiate a DDE connection to TN3270 Plus session "Mainframe"
    channel = DDEInitiate(App:="tn3270", Topic:="Mainframe")

    ' Put TN3270 Plus screen locations for the poke operations in Excel
cells A1 and A2
    ActiveCell.Range("A1") = 81"
    ActiveCell.Range("A2") = "161"

    ' Put the data to poke into the TN3270 Plus fields into Excel cells A3
and A4.
    ActiveCell.Range("A3") = "555"
    ActiveCell.Range("A4").NumberFormat = "@"       ' Format this cell as a
string.
    ActiveCell.Range("A4") = "0005093075236054321"

    ' Save the updates to cells A1 - A4
    ActiveWorkbook.Save

    ' Move the cursor to the first field
    Set rangeToPoke = Worksheets("Sheet1").Range("A1")
    Application.DDEPoke channel, "Cursor", rangeToPoke

    ' Write data to the first TN3270 Plus field
    Set rangeToPoke = Worksheets("Sheet1").Range("A3")
    DDEPoke channel, "Keystroke", rangeToPoke
```

```
    ' Move the cursor to the second field
    Set rangeToPoke = Worksheets("Sheet1").Range("A2")
    Application.DDEPoke channel, "Cursor", rangeToPoke

    ' Write data to the second TN3270 Plus field
    Set rangeToPoke = Worksheets("Sheet1").Range("A4")
    DDEPoke channel, "Keystroke", rangeToPoke

    ' Terminate the DDE session
    Application.DDETerminate channel

End Sub
```

6.3 Script Language

6.3.1 Script Commands (Release 3.6 and above)

6.3.1.1 Using the Scripting Language (Release 3.6 and above)

The TN3270 Plus scripting language is designed to automate common tasks and keystroke sequences. Each day the same keystrokes are entered over and over again. By creating a script, common sequences of keystrokes can be stored in a file. When the script file is run, the keystrokes are entered automatically.

Here are some common uses for scripts:

- Connecting terminal emulation sessions.
- Logging on terminal emulation users.
- Transferring files from the PC to the Host or from the Host to the PC.
- Performing a series of commonly issued commands.

A script file is a text file (.txt) that contains script commands. A script simulates the activities of a user at the keyboard. The script file can be created using your favorite word processor or text editor. Any application than can create a standard text (.txt) file will work.

Script Language Syntax Definition

The syntax descriptions of the script commands use the following notational conventions:

1. **Magenta** colored words are replaced by user input.
2. **Blue** colored symbols are part of the syntax definition and are not included in the command.
3. The following symbols are part of the command and should be entered exactly as they appear in the command format:
   , comma
   : colon
   " double quotation marks
   () Parentheses
4. Square brackets **[ ]** indicate an optional parameter and are not included in the command.
5. Braces **{ }** are used with the vertical bar **|** to indicate choices between two or more mutually exclusive items and are not included in the command.
6. Words starting with a dollar sign **$** are variables. Variable names are case sensitive.
7. Words starting in column one and ending with a colon **:** are labels. Label names are case sensitive.

8. An ellipsis **...** indicates that the same pattern continues.
9. Lines starting with an single quote (') are treated as comments.

Each line in a script file contains a command in the following format:

[lvariable =] command([parameter1[,parameter2,parameter3,...parametern]])

The command indicates the function to perform and the parameter(s) provide the data needed for the function. Parameters containing embedded blanks or commas must be enclosed in double quotation marks. Parameters may be split across multiple lines. An underscore (_) indicates more parameters on the next line. For example,

```
MsgBox("This is a test of the continuation _
        character")
```

The command is not case sensitive, so it may appear in any combination of upper and lower case. Any blanks or tabs between the command and the first parameter are ignored.

Comment lines can be added to a script file by placing an single quote followed by the comment. For example:

```
' ------------------------------------------------------------------
' These three lines are comments
' ------------------------------------------------------------------
MsgBox("Hello World")  ' This is also a comment
```

Comments cannot be included on a line that contains a continuation.

The Script Commands section contains a complete list of script commands. The syntax of each command is presented along with examples for using the command.

Creating Your First Script

The following steps will lead you through the creation of your first script:

1. Create a new text file in the TN3270 Plus directory using you favorite word processor or text editor.
2. Type the following lines:

```
' This is a comment. -- My first script file.
type("this is a test")
key(enter)
exit
```

3. Save this script as a text file named "test.txt" in the TN3270 Plus directory. Your first script is now complete.

Running Your First Script

The following steps will lead you through running your first script:

1. Connect a TN3270 Plus session.
2. Logon to your userid.
3. Open the **Host** menu and select **Run Script...** and click on the **Browse…** button.
4. In the **Select Script File** dialog box, click on "test.txt" and then click on the **Open** button.
5. "this is a test" is typed on your screen and then "entered" (pressing the enter key is simulated by the script). The result of this script is the same as if you typed "this is a test" on the keyboard and pressed the enter key.

For an example of a logon script, see the Sample Logon Script.

6.3.1.2 Script Command Summary (Release 3.6 and above)

The following is a list of script commands. For an introduction to writing a script and the script command syntax see Using the Scripting Language.

Command	Description
AskFor	display a dialog box requesting user input.
Command	issue a command in the Windows command environment.
Connect	connect to a host using a named session.
Convert	convert screen position to row and column or vice versa.
CheckList	display a dialog box containing check boxes.
Chr	convert an ANSI character code to a character.
CursorTo	move the cursor.
DDE	perform a complete DDE conversation in a single command.
DDEExecute	send a command to a DDE server application.
DDEInitiate	begin a DDE conversation with a DDE server application.
DDEPoke	send text to a DDE server application.
DDERequest	request text from a DDE server application.
DDETerminate	close a DDE channel.
EditSelect	select data for cut and copy.
Exit	exit the script.
FileSpec	specify the file to be used in a HostSave, HostPrintScreen, HostEnableLogging or EditSaveClipboard command.
FileStat	returns the status from an open request to a disk file.
FileTransfer	initiate a file transfer.
Find	find a substring inside a string.
GetField	extract a field from a field-separated string.
GetString	read a string from a file into a variable.
Global	define a global variable.
Goto	branch to a label in the script file.
If	branch to a label in the script file based on the results of a comparison.
Include	call another script file.
Int	returns the integer portion of a decimal number.
Key	simulate a function key.
LCase	convert a string to lower case.
Left	return characters from the left side of a string.
Len	return the length of a string.
Mid	return characters from the middle of a string.
MsgBox	display a dialog box.
Option	set script options.
PutString	write a string to a file.
Replace	replace a character or substring in a string.
Right	return characters from the right side of a string.

Round	round a number.
Run	run an application.
RunDirectory	specify the working directory for a program started by the "run" command.
Session	switch to another active session.
SetEnv	set the specified value into a local environment variable.
SetPrt	change the session printer.
SetUsertEnv	set the specified value into a user environment variable.
SSLConnect	begin a TLS connection.
TextBox	display a dialog box.
Trim	trim leading and/or trailing characters from a variable.
Type	"type" characters into the session terminal.
UCase	convert a string to upper case.
Wait	wait for the specified number of milliseconds.
WaitFor	wait until the specified text string appears in the host session (WaitFor(USERID)). Or, wait until one of several specified text strings appears in the host session. When one of the text strings is found, branch to the specified label.

See Also:

Script Variables
Using the Scripting Language

6.3.1.3 AskFor

```
[$variable = ]AskFor("prompt"[,password][,"defaultData"]
             [,BackButton(label:)][,WindowPos(CENTER)])
                                  [,WindowPos(CENTRE)]
                                  [,WindowPos(DEFAULT)]
                                  [,WindowPos(CURSOR)]
                                  [,WindowPos(top,left[,PIXELS])]
                                  [,WindowPos(row,col,ROWCOL)]
                                  [,WindowPos(row,col,CURSOR)]
```

Where:

$variable	an optional return variable. If a return variable is specified, the data the user types into the **AskFor** dialog box is assigned to this variable instead of typed on the screen. If the variable already contains data and no default data is specified on the **AskFor** command; the variable data appears as the default data in the edit box within the AskFor dialog box.
prompt	is text telling the user what to enter into the **AskFor** dialog box.
password	an optional keyword that indicates asterisks (*) are displayed rather than the text the user types into the AskFor dialog box.
defaultData	this text appears in the edit box within the **AskFor** dialog box. The user can accept this data by pressing enter or change it by typing over it.
BackButton(label:)	If this parameter is specified the "Paste" button on the AskFor prompt dialog box is replaced with a "Back" button. If the "Back" button is pressed, the script branches to the label specified in the parameter instead of processing the data in the AskFor dialog.
WindowPos(CENTER)	centers the **AskFor** dialog box over the TN3270 Plus terminal window.
WindowPos(CENTRE)	centers the **AskFor** dialog box over the TN3270 Plus terminal window.
WindowPos(DEFAULT)	centers the **AskFor** dialog box over the TN3270 Plus terminal window. This is the default if the WindowPos parameter is not specified.

WindowPos(CURSOR)	positions the top left corner of the **AskFor** dialog box at the cursor location.
WindowPos(top,left[,PIXELS])	positions the top left corner of the **AskFor** dialog box at the specified pixel offset from the top left corner of the TN3270 Plus window. top and left are the offset in pixels. The offset may be negative to move the dialog box outside the TN3270 Plus window.
WindowPos(row,col ,ROWCOL)	positions the top left corner of the **AskFor** dialog box at the specified row and column in the TN3270 Plus terminal window.
WindowPos(row,col ,CURSOR)	positions the top left corner of the **AskFor** dialog box at the specified row and column offset from the cursor location in the TN3270 Plus terminal window.

prompt, defaultData, label, top, left, row and col may be variables.

The **AskFor** command displays a dialog box requesting information from the user. The **AskFor** dialog box displays the message and the user accepts the defaultData or types the requested data into the edit box. When the user clicks the **OK** button the characters from the edit box are typed into the session or placed in the optional $variable. The **AskFor** command is a good way to provide additional information to lead inexperienced users through a process.

Examples

```
AskFor("Enter your userid and then click OK",John)
AskFor("Enter your last name and click OK",Doe)
AskFor("Enter your name",BackButton(START:))
$result = AskFor("Enter your password",password)
$name = AskFor("Enter your name","John Doe",WindowPos(10,10,ROWCOL))
```

Sample Script

```
'This script prompts the user for logon information.

Askfor("Enter your userid and click OK")
key(tab)
AskFor("Enter your password and click OK",password)
key(enter)
exit
```

See Also:

MsgBox
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.4 Command

`command(command)`

Where:

command	is a command to be issued in the Windows command environment. Enclose the command in double quotation marks ("") if it contains embedded spaces, tabs or commas. This parameter may be a variable containing the command to be issued.

The **command** script command allows Windows commands to be executed from within a script.

Examples

```
command("rename test.txt test1.txt")
command("delete c:\test1.txt")
command($COMMAND)
```

See Also:

Run
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.5 Connect

```
Connect("session name")
```

Where:

session name	is the name of a session. session names are case sensitive. Enclose the session name in double quotation marks ("") if it contains embedded spaces or commas. This parameter may be a variable containing the session name.

The **Connect** command connects the specified session. Session names can be created and saved in the **Connect to Host** dialog box. The Session Name group box in the **Connect to Host** dialog box contains a drop-down list of the named sessions. Open the **Host** menu and select **Connect...** to display the **Connect to Host** dialog box. The **Connect** command starts the new session using the next available session number.

Examples

```
Connect("Host 1")
Connect("Host 2")
Connect(Host_2)
Connect($SESSION)
```

Sample Script

```
' This script connects three sessions.
'
Connect("zSeries")
Connect("iSeries")
Connect("unix")
exit
```

See Also:

Script Command Summary
Script Variables
SSLConnect
Using the Scripting Language

6.3.1.6 Convert

```
$return = Convert(position,{ROW | COL})
$return = Convert(row,column,POS)
```

Where:

position	is the screen position relative to one.
row	is a row number.
column	is a column number.

position, row and/or column may be variables.

The **Convert** command converts a screen position to a row or column, or it converts a row and column to a screen position. The **Convert** command assigns the result of the conversion to the return variable. The upper left-hand corner of the screen is position 1 or row 1 column 1. The lower right-hand corner of a 24 by 80 screen is position 1920 or row 24 column 80.

Examples

```
$row = Convert(81,ROW)
$col = Convert(81,COL)
$position = Convert($row,$col,POS)
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.7 CheckList

```
CheckList("prompt","option1",$var1,"option2",$var2...[,BackButton(label:)][
,WindowPos(CENTER)])
                                                                    [
,WindowPos(CENTRE)]
                                                                    [
,WindowPos(DEFAULT)]
                                                                    [
,WindowPos(CURSOR)]
                                                                    [
,WindowPos(top,left[,PIXELS])]
                                                                    [
,WindowPos(row,col,ROWCOL)]
                                                                    [
,WindowPos(row,col,CURSOR)]
```

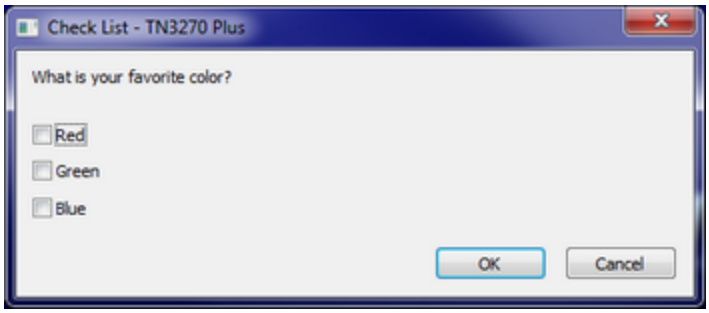
Where:

prompt	is text instructing the user to take action.
option1	is the label for the first check box.
$var1	is the variable assigned to the first check box.
option2	is the label for the second check box.
$var2	is the variable assigned to second check box.
BackButton(label:)	Displays a "Back" button in the **Check List** dialog box. Clicking the back button will cause the script to branch to the specified label.
WindowPos(CENTER)	centers the **Check List** dialog box over the TN3270 Plus terminal window.
WindowPos(CENTRE)	centers the **Check List** dialog box over the TN3270 Plus terminal window.
WindowPos(DEFAULT)	centers the **Check List** dialog box over the TN3270 Plus terminal window. This is the default if the WindowPos parameter is not specified.
WindowPos(CURSOR)	positions the top left corner of the **Check List** dialog box at the cursor location.
WindowPos(top,left[,PIXELS])	positions the top left corner of the **Check List** dialog box at the specified pixel offset from the top left corner of the TN3270 Plus window. top and left are the offset in pixels. The offset may be negative to move the dialog box outside the TN3270 Plus window.
WindowPos(row,col ,ROWCOL)	positions the top left corner of the **Check List** dialog box at the specified row and column in the TN3270 Plus terminal window.
WindowPos(row,col ,CURSOR)	positions the top left corner of the **Check List** dialog box at the specified row and column offset from the cursor location in the TN3270 Plus terminal window.

The **CheckList** command presents a list of check box options in a dialog box. The user can check the desired option(s). The variable associated with each check box contains 1 if the option is checked and 0 if the option is not checked. A variable can be set to 1 prior to the **CheckList** command to pre-set the check box.

Example:

```
CheckList("What is your favorite color?", _
        "Red",$Red, _
        "Green",$Green, _
        "Blue",$Blue)
```



Sample Script

```
' CheckList Sample
CheckList("What is your favorite color?", _
        "Red",$Red, _
        "Green",$Green, _
        "Blue",$Blue)

if $Red = 1   then $color = Red
if $Green = 1 then $color = Green
if $Blue = 1  then $color = Blue

MsgBox("Your favorite color is: " & $color)
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.8 Chr

```
$character = Chr(charcode)
```

Where:

$character	is the ANSI character represented by the character code: charcode.
charcode	is an ANSI character code.

charcode may be a variable.

The **Chr** command converts an ANSI character code to a character. For example, Chr(10) returns a line feed character. You can use Chr(10) to format multiple lines in a message box.

Examples

```
$TAB = Chr(9)
$LINEFEED = Chr(10)
```

```
MsgBox("Line 1" & Chr(10) & "Line 2")
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.9 CursorTo

```
CursorTo(position)
CursorTo(row,column)
```

Where:

position	is the screen position relative to one.
row	is a row number.
column	is a column number.

position, row and/or column may be variables.

The **CursorTo** command moves the cursor to the specified position or the specified row and column. The upper left-hand corner of the screen is position 1 or row 1 column 1. The lower right-hand corner of a 24 by 80 screen is position 1920 or row 24 column 80.

Examples

```
CursorTo(1)
CursorTo(81)
CursorTo(1,1)
CursorTo(24,80)
CursorTo($POSITION)
CursorTo($ROW,$COLUMN)
```

Restrictions:

CursorTo is not supported for ANSI and VTxxx terminal emulation.

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.10 DateAdd

```
$return = DateAdd({D|W|M|Y},interval,date[,date_format])
```

Where:

$return	The date adjusted by the specified interval.
D	The interval parameter is the number of days.
W	The interval parameter is the number of weeks.
M	The interval parameter is the number of months.
Y	The interval parameter is the number of years.
interval	The number of days, weeks, months or years to add to the date. A positive integer moves the date forward. A negative integer moves the date backward.

date	The date to be adjusted.
date_format	can be one of the following: DMY ($return and date are in DD/MM/YYYY format.) MDY ($return and date are in MM/DD/YYYY format.) YMD ($return and date are in YYYY/MM/DD or YYYY-MM-DD format.) date_format may be omitted, in which case the date format is taken from the "Option DateFormat" command. If the "Option DateFormat" command is also omitted, then the date format is assumed to be the system short date format.

interval, date and/or date_format may be variables.

The **DateAdd** command adjusts the input date by the specified interval. A positive interval moves the date forward and a negative interval moves the date backward.

Examples

```
$NewDate = DateAdd(d, 1, "2014-01-01", ymd)    ' Returns 2014-01-02
$NewDate = DateAdd(D, -1, "2014-01-01", YMD)   ' Returns 2013-12-31
$NewDate = DateAdd(d, 1, "01/01/2014", mdy)    ' Returns 01/02/2014
$NewDate = DateAdd(D, -1, "01/01/2014", dmy)   ' Returns 31/12/2013
$Tomorrow = DateAdd(d, 1, $DATEYMD, ymd)       ' Returns tomorrow's date
$Yesterday= DateAdd(d, -1, $DATEYMD, ymd)      ' Returns yesterday's date
$NextWeek = DateAdd(w, 1, $DATEYMD, ymd)       ' Returns next week's date
$NewMonth = DateAdd(m, 1, $DATEYMD, ymd)       ' Returns next month's
date
$NextYear = DateAdd(y, 1, $DATEYMD, ymd)       ' Returns next year's date
```

Sample Script

```
' DateAdd - Adjust today's date
$LineFeed = Chr(10)
$DateFormat = ymd
$Today  = $DATEYMD
$Tomorrow =  DateAdd(d, 1, $Today, $DateFormat)
$Yesterday = DateAdd(d,-1, $Today, $DateFormat)
$NextWeek =  DateAdd(w, 1, $Today, $DateFormat)
$NewMonth =  DateAdd(m, 1, $Today, $DateFormat)
$NextYear =  DateAdd(y, 1, $Today, $DateFormat)
MsgBox("Tomorrow  = " & $Tomorrow  & $LineFeed _
     & "Yesterday = " & $Yesterday & $LineFeed _
     & "Next Week = " & $NextWeek  & $LineFeed _
     & "Next Year = " & $NextYear)
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.11 Day

```
$return = Day(date[,date_format])
```

Where:

$return	A number from 1 to 31 representing the day of the month.
date	The input date.

date_format	can be one of the following: DMY (date is in DD/MM/YYYY format.) MDY (date is in MM/DD/YYYY format.) YMD (date is in YYYY/MM/DD or YYYY-MM-DD format.) date_format may be omitted, in which case the date format is taken from the "Option DateFormat" command. If the "Option DateFormat" command is also omitted, then the date format is assumed to be the system short date format.

date and date_format may be a variables.

The **Day** command returns a number between 1 and 31 for the day of the month of the input date.

Examples

```
$DayOfMonth = Day($DATE)                    "Returns today's day of
                                            the month using system
                                            short date format.
$DayOfMonth = Day("2015-01-01", YMD)        'Returns 01
$DayOfMonth = Day("2015-01-02", ymd)        'Returns 02
```

Sample Script

```
' Display the date.
$Today = $DATE
$DayNumber = Day($Today)
$MonNumber = Month($Today)
$Year = Year($Today)
'Remove leading zero
$DayNumber = Trim($DayNumber,"0",left)
$MonNumber = Trim($MonNumber,"0",left)
MsgBox("Today is day "  & $DayNumber _
         & " of month " & $MonNumber _
         & " of year "  & $Year)
exit
```

See Also:

DateAdd
Day
Month
MonthName
Weekday
WeekdayName
Year

6.3.1.12 DDE

```
DDE({POKE | EXECUTE},service,topic,item[,data][,onErrorLabel:])
$return = DDE(REQUEST,service,topic,item[,onErrorLabel:])
```

Where:

$return	is a variable that receives the result of a REQUEST operation.
POKE	sends text to the DDE server application.
REQUEST	requests text from the DDE server application.
EXECUTE	sends a command to the DDE server application.
service	is name of the DDE application for this conversation.
topic	is name of the DDE topic for this conversation.
item	is the name of the DDE item for this conversation.
data	is a string of text to be sent to the DDE server application for a POKE operation.

onErrorLabel: is a label that execution transfers to if an error occurs.

The DDE command is a shorthand command that performs a complete DDE conversation in a single command. It performs the functions of DDEInitiate, one of DDEPoke, DDERequest or DDEExecute, and DDETerminate. Use this command if you want to perform a single DDE function.

Examples

```
$result = DDE(REQUEST,"EXCEL","Sheet2",R1C1)
```

Sample Script

```
' This script starts Microsoft Excel with a test spreadsheet,
' gets the data in row 1 column 1 of the spreadsheet and types it into
' the terminal emulation screen at the current cursor location.
'
Run("C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls""")
Wait(1000)
$result = DDE(REQUEST,"EXCEL","Sheet2",R1C1)
type($result)
exit
```

See Also:

DDEExecute
DDEInitiate
DDEPoke
DDETerminate

6.3.1.13 DDEExecute

```
DDEExecute($variable,item[,onErrorLabel:])
```

Where:

$variable	is the variable containing the DDE channel number. The DDE channel is established by the DDEInitiate command.
item	is the command sent to the DDE server application.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDEExecute command sends a command to a Dynamic Data Exchange (DDE) server application.

Example

```
DDEExecute($channel,"[SAVE()][QUIT()]")
```

Sample Script

```
' This script starts Microsoft Excel with a test spreadsheet.
' "some data" is then put into row 1 column 1 of the spreadsheet.
' The spreadsheet is saved and Excel is closed.
' The DDE conversation is terminated.
'
Run("C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls""")
Wait(1000)
$channel = DDEInitiate("EXCEL","Sheet2")
DDEPoke($channel,"R1C1","some data")
DDEExecute($channel,"[SAVE()][QUIT()]")
DDETerminate($channel)
```

```
exit
```

See Also:

DDE
DDEInitiate
DDEPoke
DDERequest
DDETerminate

6.3.1.14 DDEInitiate

```
$return = DDEInitiate(service,topic[,onErrorLabel:])
```

Where:

$return	is the variable that receives the DDE channel number. Subsequent DDE script commands reference the channel number created by DDEInitiate.
service	is the name of the DDE application for this conversation.
topic	is the name of the DDE topic for this conversation.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDEInitiate command begins a Dynamic Data Exchange (DDE) conversation between TN3270 Plus and another application. The DDE conversation allows data to be exchanged between TN3270 Plus and the other application.

Example

```
$channel = DDEInitiate("EXCEL","Sheet2")
```

Sample Script

```
' This script starts Microsoft Excel with a test spreadsheet.
' "some data" is then put into row 1 column 1 of the spreadsheet.
' The spreadsheet is saved and Excel is closed.
' The DDE conversation is terminated.
'
Run("C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls""")
Wait(1000)
$channel = DDEInitiate("EXCEL","Sheet2")
DDEPoke($channel,"R1C1","some data")
DDEExecute($channel,"[SAVE()][QUIT()]")
DDETerminate($channel)
exit
```

See Also:

DDE
DDEExecute
DDEPoke
DDERequest
DDETerminate

6.3.1.15 DDEPoke

```
DDEPoke($variable,item,data[,onErrorLabel:])
```

Where:

$variable	is the variable containing the DDE channel number. The DDE channel is established by the DDEInitiate command.
item	is the name of the item to be updated.
data	is a string of text to be sent to the DDE server application.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDEPoke command sends text to a Dynamic Data Exchange (DDE) server application.

Example

```
DDEPoke($channel,"R1C1","some data")
```

Sample Script

```
' This script starts Microsoft Excel with a test spreadsheet.
' "some data" is then put into row 1 column 1 of the spreadsheet.
' The spreadsheet is saved and Excel is closed.
' The DDE conversation is terminated.
'
Run("C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls""")
Wait(1000)
$channel = DDEInitiate("EXCEL","Sheet2")
DDEPoke($channel,"R1C1","some data")
DDEExecute($channel,"[SAVE()][QUIT()]")
DDETerminate($channel)
exit
```

See Also:

DDE
DDEExecute
DDEInitiate
DDERequest
DDETerminate

6.3.1.16 DDERequest

```
$return = DDERquest($variable,item[,onErrorLabel:])
```

Where:

$return	is a variable that receives the result of the request..
$variable	is the variable containing the DDE channel number. The DDE channel is established by the DDEInitiate command.
item	is the name of the item being requested.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDERequest command receives text from a Dynamic Data Exchange (DDE) server application.

Example

```
$result = DDERequest($channel,"R1C1")
```

Sample Script

```
' This script starts Microsoft Excel with a test spreadsheet,
' gets the data in row 1 column 1 of the spreadsheet, terminates Excel,
' and types the requested text into the terminal emulation screen at
```

```
' the current cursor location.
'
Run("C:\Program Files\Microsoft Office\Office11\excel.exe
         ""D:\Test.xls""")
Wait(1000)
$channel = DDEInitiate("EXCEL","Sheet2")
$result = DDERequest($channel,"R1C1")
DDEExecute($channel,"[QUIT()]")
DDETerminate($channel)
type($result)
```

See Also:

DDE
DDEExecute
DDEInitiate
DDEPoke
DDETerminate

6.3.1.17 DDETerminate

```
DDETerminate($variable)
```

Where:

$variable	is the variable containing the DDE channel number. The DDE channel is established by the DDEInitiate command.

The DDETerminate command closes the Dynamic Data Exchange (DDE) channel.

Example

```
DDETerminate($channel)
```

Sample Script

```
' This script starts Microsoft Excel with a test spreadsheet,
' gets the data in row 1 column 1 of the spreadsheet, terminates Excel,
' and types the requested text into the terminal emulation screen at
' the current cursor location.
'
Run("C:\Program Files\Microsoft Office\Office11\excel.exe
         ""D:\Test.xls""")
Wait(1000)
$channel = DDEInitiate("EXCEL","Sheet2")
$result = DDERequest($channel,"R1C1")
DDEExecute($channel,"[QUIT()]")
DDETerminate($channel)
type($result)
```

See Also:

DDEInitiate
DDEExecute
DDEPoke
DDERequest

6.3.1.18 EditSelect

```
EditSelect(top,left,bottom,right)
```

Where:

top	is the first row of the selection rectangle.
left	is the left column of the selection rectangle.
bottom	is the last row of the selection rectangle.
right	is the right column of the selection rectangle.

top, left, bottom and/or right may be variables.

The **EditSelect** command selects data in the specified rectangle. The selected data may be moved to the clipboard using the following commands:

```
Key(EditCopy)
key(EditCopyAppend)
Key(EditCut)
Key(EditCutAppend)
```

Examples

```
EditSelect(1,1,10,80)
EditSelect(24,1,24,80)
EditSelect($TOP,$LEFT,$BOTTOM,$RIGHT)
```

Sample Script

```
' This script copies screen data to the clipboard
' and then saves the data to a file.
EditSelect(1,1,5,80)
key(EditCopy)
FileSpec(clipboard,c:\ClipboardData.txt",append)
key(EditSaveClipboard)
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.19 Exit

```
exit
```

The **exit** command terminates a script. The exit command has no parameters.

Example

```
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.20 FileSpec

```
FileSpec({log | printer | clipboard}, "filename"[,append])
```

Where:

log	indicates a log file specification for the HostSave or HostEnableLogging system commands.
printer	indicates a printer file specification for the HostPrintScreen system command.
clipboard	indicates a file specification for the EditSaveClipboard system command.
filename	is the full filename, including drive letter and path name for the output file.
append	append output to an existing file.

filename may be a variable.

The **FileSpec** command defines the disk output files for TN3270 Plus system commands.

The "FileSpec log" command directs the terminal session activity to the specified output file when logging is enabled (HostEnableLogging). The HostSave system command output is also directed to the specified file. The HostSave system command can be issued with the script command "key(HostSave)".

The "FileSpec printer" command directs the PrintScreen system command output to a disk file. The HostPrintScreen system command can be issued with the script command "key(HostPrintScreen)". To reset output to the printer, use the following:

```
FileSpec(printer,"")
```

The "FileSpec clipboard" command directs the EditSaveClipboard system command output to the specified file The EditSaveClipboard command can be issued with the script command "key(EditSaveClipboard)."

Examples

```
FileSpec(log,"c:\test.log",append)
FileSpec(log,$FILENAME)
FileSpec(printer,"c:\printer.txt")
FileSpec(clipboard,"d:\clipboard.txt",append)
```

Sample Script

```
' This script defines c:\test.log" as the current log file
' and then saves the current screen to the log file.
'
FileSpec(log,"c:\test.log")
key(HostSave)
exit
```

See Also:

Script Command Summary

Script Variables

Using the Scripting Language

6.3.1.21 FileStat

```
$return = FileStat(filename,mode)
```

Where:

$return	is the integer return code for one of the following standard Windows file exceptions: 0 = none. No error occurred. 1 = genericException. An unspecified error occurred. 2 = fileNotFound. The file could not be located. 3 = badPath. All or part of the path is invalid. 4 = tooManyOpenFiles. The permitted number of open files was exceeded.

5 = accessDenied. The file could not be accessed.
6 = invalidFile. There was an attempt to use an invalid file handle.
7 = removeCurrentDir. The current working directory cannot be removed.
8 = directoryFull. There are no more directory entries.
9 = badSeek. There was an error trying to set the file pointer.
10 = hardIO. There was a hardware error.
11 = sharingViolation. SHARE.EXE was not loaded, or a shared region was locked.
12 = lockViolation. There was an attempt to lock a region that was already locked.
13 = diskFull. The disk is full.
14 = endOfFile. The end of file was reached.

filename	is the full filename, including drive letter and path name for the file.
mode	is the mode of the open request. It must be one of the following. modeRead modeWrite modeReadWrite

path and mode may be a variables.

The **FileStat** command returns the status from an open request to a disk file.

Examples

```
If FileStat("D:\test.txt",modeRead) = 2 Then GoTo FileNotFound:
If FileStat("D:\test.txt",modeWrite) = 11 Then GoTo FileAlreadyOpen:
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.22 FileTransfer

PC to Host:

```
FileTransfer      Operation={send | $variable},
      PcFile={PC filename | $variable},
      HostFile={Host filename | $variable},
      Opsys={vm/cms | cics | mvs/tso | $variable},
      [Blksize={blksize | $variable},]
      [Lrecl={lrecl | $variable},]
      [Recfm={default | fixed | variable | undefined | $variable},]
      [Mode={WSF | ScreenImage | $variable},]
      [BufferSize={ buffsize | $variable},]
      [Language={language | $variable},]
      [Options={options | $variable},]
      [Program={program | $variable}]
```

Host to PC:

```
FileTransfer      Operation={receive | $variable},
      PcFile={PC filename | $variable},
      HostFile={Host filename | $variable},
```

```
Opsys={vm/cms | cics | mvs/tso | $variable},
PCOptions={replace | append | prompt | $variable},
[Mode={WSF | ScreenImage | $variable},]
[BufferSize={ buffsize | $variable},]
[Language={language | $variable},]
[Options= {options | $variable},]
[Program={program | $variable}]
```

Where:

$variable	is a variable. The text defined for the variable is substituted for the variable name. A variable may be used in place of any of the FileTransfer command keyword parameters.
send	send a file from the PC to the session host computer (upload). This parameter is not case sensitive.
receive	move a file from the session host computer to the PC (download). This parameter is not case sensitive.
PC filename	is the full filename, including drive letter and path, of a PC file. Enclose the PC filename in double quotation marks ("") if it contains embedded spaces or commas.
Host filename	is the name of a file on the host computer. Enclose the Host filename in double quotation marks ("") if it contains embedded spaces or commas.
vm/cms | cics | mvs/tso	is the operating system on the host computer. Specify "vm/cms", "cics" or "mvs/tso."
blksize	is the block size for the transferred file. A valid blksize is between 1 and 32760. (MVS/TSO send operations only)
lrecl	is the logical record length for the transferred file. A valid lrecl is between 1 and 32760. (Send operations only)
default | fixed | variable | undefined	is the record format for the host file. Specify one of the following: "default", "fixed", "variable" or "undefined". (Send operations only)
	default — use the default record format for the host system. fixed — indicates fixed length records. variable — indicates variable length records. undefined — indicates the record format is not defined.
replace | append | prompt	is one of the following options: "replace", "append" or "prompt". (Receive operations only)
	replace — if the PC file already exists, replace it with the download file. append — if the PC file already exists, add the download file to the end of the existing file. prompt — if the PC file already exists, prompt the user to replace the file or cancel the transfer operation.
WSF | ScreenImage	is the type of file transfer. WSF indicates a Write Structured Field file transfer with the buffer size specified in the BufferSize parameter. Screen images indicates a screen image file transfer. The default is WSF.
buffsize	is the buffer size for WSF file transfers. A valid buffsize is between 512 and 65535. The default is 2048.
language	specify the name of the host language code page (*.cpg) to use for this file transfer operation. If a language file is not specified, the host language code page defined for the session is used by default.
options	specify other IND$FILE options. The options in this parameter are passed directly to IND$FILE exactly as they appear. There is no editing of the options in this field. This

allows you to pass options specific to your version of IND$FILE. Enclose the options string in double quotation marks ("") if it contains embedded spaces or commas. Some common options are:

ascii	Converts ASCII to EBCDIC when the PC sends a file to the host, and converts EBCDIC to ASCII when the PC receives a file from the host.
crlf	Replaces the carriage return/line feed characters with line breaks when the PC sends a file to the host, and replaces line breaks with carriage return/line feed characters when the PC receives a file from the host.
append	Appends the transferred file to the host file. (Send operations only.)
notrunc	(z/OS (MVS) and VM) Do not truncate trailing blanks.
blank	(CICS) Do not truncate trailing blanks.

z/OS (MVS) file transfers

space(pri [,sec]) tracks | cylinders | avblock(size)
specify the space allocation for a new data set (Send only). Where:
pri is the primary allocation.
sec is the secondary allocation.
size is the number of bytes in an avblock.
tracks | cylinders | avblock(size) indicates the unit of allocation.

Examples:

space(15,1) tracks
space(2,1) cylinders
space(15,1) avblock(1024)

VSE file transfers

file=rdr | lst | pun	specify the POWER queue location for the host file. (VSE file transfers only)
file=ts	specify the host file location is in CICS/VSE temporary storage. (VSE file transfers only)
file=lib	specify the host file is in a VSE library. (VSE file transfers only)
l=libname	specify the name of the VSE library. (VSE file transfers only)
s=sublib	specify the name of the VSE sublibrary. (VSE file transfers only)

program — specify the file transfer program name. By default the program name is IND$FILE.

The **FileTransfer** command initiates a file transfer. File transfer operations are only valid for 3270 sessions.

Examples

```
FileTransfer   operation=send,
         pcfile=e:\download\testfile.txt,
         hostfile="testfile upload a",
         opsys=vm/cms,
         lrecl=80,
         recfm=fixed,
         options="ascii crlf"

FileTransfer   operation=receive,
         pcfile=e:\download\testfile.txt,
         hostfile="testfile upload a",
         opsys=vm/cms,
         pcoptions=prompt, o
         options="ascii crlf"

FileTransfer   operation=receive,
         pcfile=e:\test.txt,
         hostfile="test file a",
         opsys=vm/cms,
```

```
            pcoptions=prompt,
            language="C:\Program Files\SDI\TN3270 Plus\france.cpg",
            options="ascii crlf"

   FileTransfer   operation=send,
            pcfile=e:\test.txt,
            hostfile='APPL.TEST.DSN(TEST)',
            opsys=mvs/tso,
            options="ascii crlf"

   FileTransfer   operation=receive,
            pcfile=e:\test.txt,
            hostfile='APPL.TEST.DSN(TEST)',
            opsys=mvs/tso,
            mode=wsf, buffersize=65535,
            pcoptions=prompt,
            options="ascii crlf"
```

The following script transfers file "profile exec a" from the VM/CMS host to "C:\download\profile.txt" on the PC.

```
   FileTransfer   operation=receive,
            pcfile=c:\download\profile.txt,
            hostfile="profile exec a",
            opsys=vm/cms, pcoptions=prompt,
            options="ascii crlf"
   exit
```

The following script sends file "C:\download\profile.txt" on the PC to "test exec a" on the VM/CMS host.

```
   FileTransfer   operation=send,
            pcfile=c:\download\profile.txt,
            hostfile="test exec a",
            opsys=vm/cms,
            lrecl=80,
            recfm=default, options="ascii crlf"
   exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.23 Find

```
$return = find(stringToSearch,stringToFind[,start][,USEWILDCARDS])
```

Where:

$return	receives the position, relative to 1, where stringToFind first appears in stringToSearch . If the string is not found, "0" (zero) is returned.
stringToSearch	the string to be searched. Enclose stringToSearch in double quotation marks ("") if it contains embedded spaces or commas.
stringToFind	the string to find in stringToSearch. Enclose stringToFind in double quotation marks ("") if it contains embedded spaces or commas. When the "USEWILDCARDS" parameter is specified, this parameter may contain wildcards and a pattern matching search is done. The following wildcards may be used:

Wildcard Characters	Matches in the search string

?	Any single character
*	Zero or more characters
#	Any single numeric digit
[charlist]	Any single character in charlist
[!charlist]	Any single character not in charlist

charlist is a group of one or more characters enclosed in brackets ([]) and can be used to match any single character in a string. It may include any of the following characters:

- Almost any character in the ANSI character set including digits
- The special characters opening bracket ([), question mark (?), number sign (#) and asterisk (*) can be used for matching only if enclosed in brackets. The closing bracket (]) can't be used within a charlist to match itself, but it can be used outside the group as an individual character. A hyphen (-) can be used at the beginning or end of charlist to match itself. In any other position it indicates a range of characters.
- A range of characters can be specified by placing a hyphen between lower and upper bounds. For example, [a-z], [A-Z] or [0-9]. Multiple ranges can be specified without any delimiter. For example [a-zA-Z0-9] matches any alphanumeric character. When a range of characters is specified they must appear in ascending sort order. [A-Z] is valid, but[Z-A] is not.

start	an optional parameter that specifies the position, relative to 1, within stringToSearch where the find operation should begin.
USEWILDCARDS	use pattern matching when performing the search.

stringToSearch, stringToFind and start may be variables.

The **find** command finds a string within another string and returns the position of the found string in a variable. If the string is not found, "0" (zero) is returned. If the USEWILDCARDS parameter is specified, wildcard characters may be specified in the stringToFind parameter and a pattern matching search is performed.

Examples

```
$Position = Find("The quick brown fox",fox)
$Position = Find("Doe, John 123-456-7890",###[-]###[-]####,USEWILDCARDS)
$Position = Find($string1,$string2,10)
$Position = Find($SCREEN,Login:)
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.24 GetField

```
$return = GetField(string,field#[,separator])
```

Where:

$return	receives the field read from the string.
string	is the string containing the field to be extracted.
field#	is the field number to extract.

separator	is the single character field separator. The default is a comma. If coding a comma it should be enclosed in double quotes.

string, field# and separator may be variables.

The **GetField** command extracts a field from a field-separated string and copies it into a script variable.

Examples

```
$Field = GetField("The quick brown fox",2," ")
$Field = GetField("The,quick,brown,fox",2,",")
```

Sample Script

```
' GetField.txt
' Sample script to extract fields
'
$String = "The quick brown fox"
$FieldNumber = 1
$Fields = 4

LOOP:
$Field = GetField($String,$FieldNumber," ")
type($Field)
$FieldNumber = $FieldNumber + 1
if $FieldNumber < $Fields then GoTo LOOP:
Exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.25 GetString

```
$return = GetString(filename,line#[,onErrorLabel:])
```

Where:

$return	receives the text read from the file.
filename	is the full filename, including drive letter and path name for the input file. This file must be a text file. Enclose the filename in double quotation marks ("") if it contains embedded spaces or commas.
line#	is the line number of the line to be read from the input file.
onErrorLabel:	is the label of an error handler routine. The script branches to this routine if the I/O fails. The I/O error handler routine can use the $FILEERROR variable to get the cause of the error. If this parameter is omitted, the user will be prompted with the script error dialog box which allows the command to be retried or ignored.

filename and line# may be variables.

The **GetString** command reads a line of text from a file and copies it into a script variable.

Examples

```
$Line = GetString(c:\test.txt,1)
$Line = GetString(c:\test.txt,1,IOError:)
$Line = GetString($FileName,$LineNumber)
```

Sample Script

```
' GetString.txt
' Sample script to read a file.
'
$Filename = c:\sample.txt
$LineNumber = 1

LOOP:
$Line = GetString($Filename,$LineNumber)
if $FILEERROR = 14 then GoTo EndOfFile:
if $FILEERROR <> 0 then GoTo IOError:
$LineNumber = $LineNumber,+,1
GoTo LOOP:

EndOfFile:
exit

IOError:
MsgBox("IO error. Return code = " & $FILEERROR)
exit
```

See Also:

PutString Command
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.26 GetStringAt

```
$return = GetStringAt(string,start[,min_length,[,[\]left_delimiter[,[\]
right_delimiter]]][,LEFTWILDCARDS | RIGHTWILDCARDS | USEWILDCARDS] )
```

Where:

$return	the delimited string extracted from the input string.
string	the input string.
start	the starting position within the input string.
min_length	the search for the right delimiter begins at the starting position plus the min_length. The default min_length is 1.
left_delimiter:	the left delimiter. The left delimiter can be a single character delimiter, a string of single character delimiters or a string delimiter. If a string of single characters is used, any one of the characters will act as the delimiter. To identify the entire string as the delimiter, you precede the string with a backslash. Enclose the left_delimiter in double quotation marks ("") if it contains embedded spaces or commas. If you are using a backslash to indicate a delimiter string and the string must be enclosed in double quotes, the backslash must be the first character following the opening double quote. The default left delimiter is a space. When the "LEFTWILDCARDS" or "USEWILDCARDS" parameter is specified, this parameter may contain wildcards and a pattern matching search is done for the delimiter.
right_delimiter:	the right delimiter string. The right delimiter can be a single character delimiter, a string of single character delimiters or a string delimiter. If a string of single characters is used, any one of the characters will act as the delimiter. To identify the entire string as the delimiter, you precede the string with a backslash. Enclose the right_delimiter in double quotation marks ("") if it contains embedded spaces or commas. If you are using a backslash to indicate a delimiter string and the string must be enclosed in double quotes, the backslash must be the first character following the opening double quote. The default right delimiter is a space. When the "RIGHTWILDCARDS" or "USEWILDCARDS" parameter is specified, this parameter may contain wildcards and a pattern matching search is done for the delimiter.

LEFTWILDCARDS	use pattern matching for the left_delimiter.
RIGHTWILDCARDS	use pattern matching for the right_delimiter.
USEWILDCARDS	use pattern matching for both the left_delimiter and the right_delimiter.

The following wildcards may be used for pattern matching in the delimiters:

Wildcard Characters	Matches in the search string
?	Any single character
*	Zero or more characters
#	Any single numeric digit
[charlist]	Any single character in charlist
[!charlist]	Any single character not in charlist

charlist is a group of one or more characters enclosed in brackets ([]) and can be used to match any single character in a string. It may include any of the following characters:

- Almost any character in the ANSI character set including digits
- The special characters opening bracket ([), question mark (?), number sign (#) and asterisk (*) can be used for matching only if enclosed in brackets. The closing bracket (]) can't be used within a charlist to match itself, but it can be used outside the group as an individual character. A hyphen (-) can be used at the beginning or end of charlist to match itself. In any other position it indicates a range of characters.
- A range of characters can be specified by placing a hyphen between lower and upper bounds. For example, [a-z], [A-Z] or [0-9]. Multiple ranges can be specified without any delimiter. For example [a-zA-Z0-9] matches any alphanumeric character. When a range of characters is specified they must appear in ascending sort order. [A-Z] is valid, but[Z-A] is not.

string, start, min_length, left_delimiter and right_delimiter may be variables.

The **GetStringAt** command extracts a delimited string from an input string. If the start position is greater than zero, the input string is searched backwards for the left delimiter. If the start position is zero, the search for the left delimiter starts at the beginning and the input string is searched forward for the left delimiter. If the left delimiter is found, the command extracts characters starting at the first character following the left delimiter character or string. If the left delimiter is not found, the text from the input string is extracted starting at the first character. Extraction continues until the right delimiter is found. If the right delimiter is not found, the text from the input string is extracted until the end of string is reached. If neither delimiter is found, the entire string is returned.

Examples

```
$Word = GetStringAt("Every good boy does fine.",7,4)
$DollarAmount = GetStringAt("$100.00",1,1,"$",".")
$Quantity = GetStringAt("Sample size: 500. ",14,1,": ",". ")
$Words = GetStringAt("Every good boy does fine.",1,1,"\Every ","\ fine")
```

Sample Script

```
' GetStringAt.txt
' Sample script to display "good boy"
'
```

```
$Sentence = "Every good boy does fine"
$Start = Find($Sentence, "good")
$Result = GetStringAt($Sentence,$Start,5)
Msgbox($Result)
exit
```

See Also:

Left
Mid
Right
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.27 Global

```
global($var1[,$var2] ... [,$varn])
```

Where:

$varn	is a variable. The variable is available to scripts called by using the include script command.

The **global** command defines global variables. The data in global variables is available to scripts called using the include script command. Local variables are not available to scripts called using the include command.

Examples

```
global($NAME)
global($USERID,$PASSWORD)
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.28 GoTo

```
goto  label:
```

Where:

label	is a label in the script. label names are case sensitive, contain no embedded blanks and end with a colon. This parameter may be a variable containing a label name.

The **goto** command branches to the specified label. Labels are defined by placing a label name on a script line starting in column one.

Examples

```
goto TEST_LABEL:
goto LABEL1:
goto $LABELNAME
```

Sample Script

```
' The script prompts the user for the answer to 2 + 2.
' If the user response is incorrect, the prompt is repeated.
'
Retry:
```

```
$response = AskFor("What is 2 + 2?")
if $response = 4 then goto Correct:
MsgBox("Incorrect, try again.")
goto Retry:
Correct:
MsgBox("Correct!")
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.29 If

```
if condition [AND|OR condition] then statement
```

```
if condition [AND|OR condition] then
      statement(s)
end if
```

```
if condition [AND|OR condition] then
      statement(s)
else
      statement(s)
end if
```

```
if condition [AND|OR condition] then
      statement(s)
elseIf condition [AND|OR condition] then
      statement(s)
else
      statement(s)
end if
```

Where:

condition	is an expression that evaluates to true or false. See the "Comparison Operators" in Script Operators for list of operators you can use in your If statements.
AND|OR	are keywords used to combine conditional statements in the same **IF** command. The **AND** and **OR** keywords are not case sensitive. Use **AND** when all of two or more conditions must be true. Use **OR** when one of two or more conditions must be true. Multiple **AND** and **OR** keywords may be used in a single IF statement.
statement(s)	is a script statement or statements to be executed if the condition is true. If there are multiple statements, each statement must be on a separate line.

The **If** command tests a condition or conditions and conditionally processes one or more statements based on the result.
Nested **If** commands are not supported.

Examples

```
if $name EQ "John" then type("Doe")
```

```
if $SCREEN(1,1,1,4) = "John" then
   $user = "John"
else
   $user = "Other"
end if
if $name = "John" then goto ProcessName:
if $name LIKE "J*" then goto ProceesJ:
if $number GT 100 then goto ProcessLarge:
if $a = 0 and $b = 0 then goto End:
if $a = 0 OR $b = 0 then goto End:
if $a = 0 and $b = 0 or $c = 0 then goto End:
```

See Also:

Script Command Summary
Script Operators
Script Variables
Using the Scripting Language

6.3.1.30 Include

```
include(filename)
```

Where:

filename	is the full filename, including drive letter and path name, of a script file. Enclose the script filename in double quotation marks ("") if it contains embedded spaces or commas. This parameter may be a variable containing a script filename.

The **include** command calls another script file. All the commands in the called script file are executed, and then processing continues in the calling script. An included script file may include additional script files. There is no limitation on how deeply script files can be nested.

If you have a group of commands that are used in many different scripts, put the group of commands into a separate script file and include that script file in place of the actual commands. Variables defined in the calling script are not available to the included script unless they are defined using the global script command.

Examples

```
include(c:script.txt)
include(c:\scripts\script.txt)
include($FILENAME)
```

Sample Script

```
'This script calls scripts to logon 2 users.
include("c:\scripts\logon user 1.txt")
include("c:\scripts\logon user 2.txt")
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.31 Int

```
$return = int(decimal_number)
```

Where:

$return	receives the integer portion of the decimal number.
decimal_number	is a decimal number.

decimal_number may be a variable.

The **int** command returns the integer portion of a decimal number.

Examples

```
$result = int(1.5)
$result = int($number)
```

See Also:

Round
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.32 Key

```
key({keyname | command | $variable})
```

Where:

keyname	is the name of the key on the host keyboard.
command	is the name of a system command. This parameter allows TN3270 Plus menu commands to be issued from within a script.
$variable	is a variable. The text defined for the variable is substituted for the variable name and used as the command parameter.

The **key** command specifies the key to be simulated. The complete list of keynames can be found in the **Keyboard Map Setup** dialog box. To display the **Keyboard Map Setup** dialog box, open the **Setup** menu, select **Sessions...** from **Setup Items** choose **Keyboard** and then click the **Configure...** button. Select Terminal Keys in the **Function Group** drop-down list box.

You may also consult the default keyboard maps in this help file:

Default 3270 Keyboard Map
Default 5250 Keyboard Map
Default VT100/VT220 Keyboard Map

For VT100, VT220 and ANSI terminal emulation sessions, you can also send "Ctrl+A" through "Ctrl+Z" . For example, to send "Ctrl+A" you would use the following command. Note that the "Ctrl+A" is case sensitive.

```
key(Ctrl+A)
```

The **key** command can also issue a system command from within a script. The complete list of system commands can be found in the **Keyboard Map Setup** dialog box. To display the **Keyboard Map Setup** dialog box, open the **Setup** menu, select **Sessions...** from **Setup Items** choose **Keyboard** and then click the **Configure...** button. Select System Commands in the **Function Group** drop-down list box.

Examples (3270)

```
key(enter)
key(tab)
key(PA2)
key(PF1)
key($KEY)
key(HostExit)        'Exit TN3270 Plus
```

Examples (5250)

```
key(enter)
key(tab)
key(RollDown/PageUp)
key(F1)
```

Examples (VT100/VT220)

```
key(enter)
key(tab)
key(PageUp)
key(F1)
key(Ctrl+A)
```

Examples (System Commands)

```
key(EditCopy)              'Copy selected data to the clipboard
key(EditPaste)             'Paste data into a session
key(EditSelectionLeft)     'Select the character left of the cursor
key(HostClose)             'Close the session
Key(HostEnableLogging)     'Enable terminal session activity logging
key(HostExit)              'Exit TN3270 Plus
key(HostPrintScreen)       'Print the current screen
```

Sample Script

```
'The following script types "this is a test" and then "presses" the
        enter key.
'
type("this is a test")
key(enter)
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.33 LCase

```
$return = LCase(string)
```

Where:

$return	receives the lower case string.
string	a string of characters.

string may be a variable.

The **LCase** command converts the upper case characters in the string to lower case characters. Lower case characters and numeric characters are not changed.

Examples

```
$return = lcase("This Is A Test")
$return = lcase($text)
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language
UCase

6.3.1.34 Left

```
$return = left(string,length)
```

Where:

$return	the characters from the left side of the string.
string	the input string of characters.
length	the number of characters to return. If 0 is specified, no characters are returned. If the length is greater than the input string length, the entire string is returned.

string and length may be variables.

The **Left** command returns the specified number of characters from the left side of the string.

Examples

```
$return = left("abcefgh",4)
$return = left($text,$length)
```

See Also:

Mid
Right
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.35 Len

```
$return = len(string)
```

Where:

$return	the length of the string.
string	the input string of characters.

string may be a variable.

The **Len** command returns the length of a string.

Examples

```
$return = len("abcefgh")
$return = len($name)
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.36 Mid

```
$return = mid(string,start[,length])
```

Where:

$return	the characters from the middle of the string.
string	the input string of characters.
start	the starting position within the string.
length	the number of characters to return. If length is not specified, all the characters from

the starting position to the end of the string are returned.

string, start and length may be variables.

The **mid** command returns the specified number of characters starting at the start position.

Examples

```
$return = mid("abcefgh",2)
$return = mid("abcefgh",2,2)
$return = mid($text,$start,$length)
```

See Also:

Left
Right
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.37 Month

```
$return = Month(date[,date_format])
```

Where:

$return	A number from 1 to 12 representing the month of the year.
date	The input date.
date_format	can be one of the following: DMY (date is in DD/MM/YYYY format.) MDY (date is in MM/DD/YYYY format.) YMD (date is in YYYY/MM/DD or YYYY-MM-DD format.) date_format may be omitted, in which case the date format is taken from the "Option DateFormat" command. If the "Option DateFormat" command is also omitted, then the date format is assumed to be the system short date format.

date and date_format may be a variables.

The **Month** command returns a number between 1 and 12 for the month of the year of the input date.

Examples

```
$MonthOfYear = Month($DATE)            'Returns the number of the
                                       current month in system short
                                       date format

$MonthOfYear = Month("2015-01-01",     'Returns 01
YMD)

$MonthOfYear = Month("2015-02-01",     'Returns 02
ymd)
```

Sample Script

```
' Display the date.
$Today = $DATE
$DayNumber = Day($Today)
$MonNumber = Month($Today)
$Year = Year($Today)
'Remove leading zero
```

```
$DayNumber = Trim($DayNumber,"0",left)
$MonNumber = Trim($MonNumber,"0",left)
MsgBox("Today is day "  & $DayNumber _
        & " of month " & $MonNumber _
        & " of year "  & $Year)
exit
```

See Also:

DateAdd
Day
MonthName
Weekday
WeekdayName
Year

6.3.1.38 MonthName

```
$return = MonthName(month[,abbreviate])
$return = MonthName(date[,abbreviate][,date_format])
```

Where:

$return	The month name of the specified month of the year.
month	The number of the the month of the year. See the Month script command.
date	The input date.
abbreviate	Specify "true" to return an abbreviated month name. Specify "false" to return the full month name. ("false" is the default)
date_format	can be one of the following: DMY (date is in DD/MM/YYYY format.) MDY (date is in MM/DD/YYYY format.) YMD (date is in YYYY/MM/DD or YYYY-MM-DD format.) date_format may be omitted, in which case the date format is taken from the "Option DateFormat" command. If the "Option DateFormat" command is also omitted, then the date format is assumed to be the system short date format.

month, date and date_format may be a variables.

The **MonthName** command returns the month name for the specified month of the year.

Examples

```
$MonthName = MonthName($DATE)          'Returns today's month name in
                                       system short  date format

$MonthName =                           'Returns January
MonthName("2015-01-01",YMD)

$MonthName =                           'Returns Feb
MonthName("2015-02-01",true,ymd)
```

Sample Script

```
' Display the date.
$Today = $DATE
$DayName = WeekdayName($Today)
$DayNumber = Day($Today)
$MonName = MonthName($Today)
$Year = Year($Today)
```

```
MsgBox("Today is "  & $DayName & ", " & $MonName _
                & " " & $DayNumber & ", "  & $Year)
exit
```

See Also:

DateAdd
Day
Month
Weekday
WeekdayName
Year

6.3.1.39 MsgBox

```
MsgBox("prompt"[[,icon][,"caption"]])
```

Where:

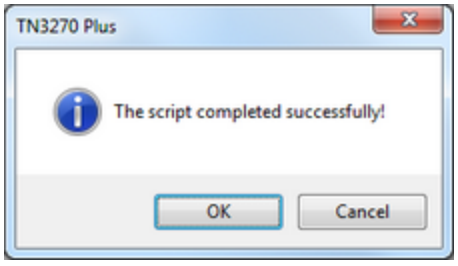
prompt	is the text to display in the message dialog box. This parameter may be a variable containing the message text.
icon	is one of the following Windows message box icons: ICONSTOP, ICONEXCLAMATION, ICONQUESTION, ICONINFORMATION. If omitted, ICONEXCLAMATION is used. Use ICONNONE to display the message box without an icon. This parameter may be a variable containing the name of the ICON to use.
caption	is the caption that appears in the title bar of the message box. If omitted the caption will be "TN3270 Plus". This parameter may be a variable containing the message box caption.

prompt, icon and caption may be variables.

The **MsgBox** command pauses the script and displays a dialog box that contains a message for the user. The **MsgBox** dialog box displays the prompt and **OK** and **Cancel** buttons. Click the OK button to continue script processing. Click the Cancel button to cancel the script.

Example

```
MsgBox("The script completed successfully!",ICONINFORMATION)
```



See Also:

Script Command Summary
Script Variables
TextBox
Using the Scripting Language

6.3.1.40 Option

```
option([option1][,option2])
```

Where:

option1	is one of the options listed below.
option2	is one of the options listed below.

Valid options are:

AskForPos(CENTER)	positions the **AskFor** dialog box in the center of the TN3270 Plus main window.
AskForPos(CENTRE)	positions the **AskFor** dialog box in the center of the TN3270 Plus main window.
AskForPos(DEFAULT)	positions the **AskFor** dialog box in the center of the TN3270 Plus main window.
AskForPos(CURSOR)	positions the top left corner of the **AskFor** dialog box at the cursor location.
AskForPos(top,left[,PIXELS])	positions the top left corner of the **AskFor** dialog box at the specified pixel offset from the top left corner of the TN3270 Plus window. top and left are the offset in pixels. The offset may be negative to move the dialog box outside the TN3270 Plus window.
AskForPos(row,col,ROWCOL)	positions the top left corner of the **AskFor** dialog box at the specified row and column in the TN3270 Plus terminal window.
AskForPos(row,col,CURSOR)	positions the top left corner of the **AskFor** dialog box at the specified row and column offset from the cursor location in the TN3270 Plus terminal window.
CheckListPos(CENTER)	positions the **CheckList** dialog box in the center of the TN3270 Plus main window.
CheckListPos(CENTRE)	positions the **CheckList** dialog box in the center of the TN3270 Plus main window.
CheckListPos(DEFAULT)	positions the **CheckList** dialog box in the center of the TN3270 Plus main window.
CheckListPos(CUSOR)	positions the top left corner of the **CheckList** dialog box at the cursor location.
CheckListPos(top,left[,PIXELS])	positions the top left corner of the **CheckList** dialog box at the specified pixel offset from the top left corner of the TN3270 Plus window. top and left are the offset in pixels. The offset may be negative to move the dialog box outside the TN3270 Plus window.
CheckListPos(row,col,ROWCOL)	positions the top left corner of the **CheckList** dialog box at the specified row and column in the TN3270 Plus terminal window.
CheckListPos(row,col,CURSOR)	positions the top left corner of the **CheckList** dialog box at the specified row and column offset from the cursor location in the TN3270 Plus terminal window.
DateFormat({YMD|DMY|MDY})	set the format for the date parameters used in script commands. YMD = yyyy/mm/dd DMY = dd/mm/yyyy MDY = mm/dd/yyyy If omitted, the date format is assumed to be the system short date format.
TextBoxPos(CENTER)	positions the **TextBox** dialog box in the center of the TN3270 Plus main window.
TextBoxPos(CENTRE)	positions the **TextBox** dialog box in the center of the TN3270 Plus main window.
TextBoxPos(DEFAULT)	positions the **TextBox** dialog box in the center of the TN3270 Plus main window.
TextBoxPos(CUSOR)	positions the top left corner of the **TextBox** dialog box at the cursor location.
TextBoxPos(top,left[,width,height][,PIXELS])	positions the top left corner of the **TextBox** dialog box at the specified pixel offset from the top left corner of the TN3270 Plus window. top and left are the offset in pixels. The offset may be negative to move the dialog box outside the TN3270 Plus window. width and height define the size of the dialog box in pixels.
TextBoxPos(top,left[,width,height],ROWCOL)	positions the top left corner of the **TextBox** dialog box at the specified row and column in the TN3270 Plus terminal window. width and height define the size of the dialog box in terminal rows and columns.
TextBoxPos(top,left[,width,height],CURSOR)	positions the top left corner of the **Text Box** dialog box at the specified row and column offset from the cursor location in the TN3270 Plus terminal window. width and height define the size of the dialog box in terminal rows and columns.

WaitForPos(CENTER)	positions the **WaitFor** dialog box in the center of the TN3270 Plus main window.
WaitForPos(CENTRE)	positions the **WaitFor** dialog box in the center of the TN3270 Plus main window.
WaitForPos(DEFAULT)	positions the **WaitFor** dialog box in the center of the TN3270 Plus main window.
WaitForPos(CURSOR)	positions the top left corner of the **WiatFor** dialog box at the cursor location.
WaitForPos(top,left[,PIXELS])	positions the top left corner of the **WaitFor** dialog box at the specified pixel offset from the top left corner of the TN3270 Plus window. top and left are the offset in pixels. The offset may be negative to move the dialog box outside the TN3270 Plus window.
WaitForPos(row,col,ROWCOL)	positions the top left corner of the **WaitFor** dialog box at the specified row and column in the TN3270 Plus terminal window.
WaitForPos(row,col,CURSOR)	positions the top left corner of the **WaitFor** dialog box at the specified row and column offset from the cursor location in the TN3270 Plus terminal window.

The **option** command specifies one or more options for a script. Options remain in effect until another option command is processed or to the end of the script.

Examples

```
option(AskForPos(50,50))
option(AskForPos(center),WaitForPos(CENTER))
option(AskForPos(0,-100))
option(DateForamt(YMD))
option(TextBoxPos(0,0,60,20,ROWCOL)
option(WaitForPos(10,10,ROWCOL))
option(WaitForPos(center))
```

6.3.1.41 PutString

```
PutString("text",filename,{create | append}[,onErrorLabel:])
```

Where:

text	is the text to write to the file.
filename	is the full filename, including drive letter and path name for the output file. This file must be a text file. Enclose the filename in double quotation marks ("") if it contains embedded spaces or commas.
create	Create a new file. If a file with the same name already exits, it is deleted.
append	Append text to an existing file. If the file does not exist, it is created.
onErrorLabel:	is the label of an error handler routine. The script branches to this routine if the I/O fails. The I/O error handler routine can use the $FILEERROR variable to get the cause of the error. If this parameter is omitted, the user will be prompted with the script error dialog box which allows the command to be retried or ignored.

text and filename may be variables.

The **PutString** command writes a line of text to a file.

Examples

```
PutString("This is a test",c:\test.txt,create)
PutString("This is a test",c:\test.txt,create,IOError:)
PutString($SCREEN(1,80),$FileName,append)
```

Sample Script

```
' PutString.txt
' This script writes the screen contents to a file.
'
$FileName = f:\script\PutStringFile.txt
$Position = 1
$Columns = 80
$MaxPosition = 1920
```

```
' Write Screen contents line by line to the file
LOOP:
PutString($SCREEN($Position,$Columns),$FileName,append)
$Position = $Position,+,$Columns
if $Position < $MaxPosition then GoTo LOOP:

Exit
```

See Also:

GetString Command
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.42 RelToAbs

```
$return = RelToAbs(position,top,left,bottom,right[,RECT])
```

Where:

$return	is the absolute screen position relative to one.
position	is the position within the selection area relative to 1.
top	is the top row of the selection area.
left	is the left column of the selection area.
bottom	is the bottom row of the selection area.
right	is the right column of the selection area.
RECT	if RECT is specified, the selection area is a rectangle with its upper left corner at top,left and its lower right corner at bottom,right. If RECT is not specified, the selection is the string of characters starting at top,left and ending at bottom,right.

position, top, left, bottom and/or right may be variables.

The **RelToAbs** command converts a position relative to the beginning of the selection area to an absolute screen position.

Examples

```
$return = RelToAbs(10,5,1,10,80)
$return = RelToAbs(15,5,1,10,10,RECT)
```

Sample Script

```
' RelToAbs.txt
' This script locates the absolute screen position of position 74 within
        the
' rectangle with its upper left corner at row 10 column 1 and its lower
        right
' corner at row 20 column 30.
$Rect = "10,1,20,30,RECT"
$Pos = RelToAbs(75,$Rect)
CursorTo($Pos)
msgbox($Pos)
Exit
```

6.3.1.43 Replace

```
$return = replace(string,oldString,newString)
```

Where:

$return	the string with the specified characters replaced..
string	variable containing text to be replaced..
oldString	the string to replaced. Enclose oldString in double quotation marks ("") if it contains embedded spaces or commas.
newString	the replacement string. Enclose newString in double quotation marks ("") if it contains embedded spaces or commas.

string, oldString and newString may be variables.

The **replace** command replaces all occurrences of oldString in string with newString.

Example

```
$return = replace($variable,dog,cat)
```

Sample Script

```
' This script replaces fox with dog

$string = "The quick brown fox."
$old = "fox"
$new = "dog"
$return = replace($string,$old,$new)
MsgBox($return)
exit
```

6.3.1.44 Right

```
$return = right(string,length)
```

Where:

$return	the characters from the right side of the string.
string	the input string of characters.
length	the number of characters to return. If 0 is specified, no characters are returned. If the length is greater than the input string length, the entire string is returned.

string and length may be variables.

The **Right** command returns the specified number of characters from the right side of the string.

Examples

```
$return = right("abcefgh",4)
$return = right($text,$length)
```

See Also:

Left
Mid
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.45 Round

```
$return = round(number[,precision][,TRIM])
```

Where:

$return	the rounded number.
number	is a decimal number.
precision	is the number of decimal places to include in the rounded result. If the precision

	specified is greater than the number of digits in the input, then trailing zeros will be added to the output.
TRIM	trims trailing zeros and trailing decimal point if nothing follows it after trimming zeros.

number and precision may be variables.

The **Round** command rounds a number to the specified precision.

Examples

```
$result = round("3.1415",2)
```

Sample Script

```
' round pi
$pi = 3.1415
$a = round($pi)
$b = round($pi,1)
$c = round($pi,10)
MsgBox($a & $CRLF & $b & $CRLF & $c & $CRLF)
exit
```

See Also:

int
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.46 Run

```
run(filename{,HIDDEN | MINIMIZED | MAXIMIZED})
```

Where:

filename	is the full file name, including drive letter and path, of an application or batch file. Enclose the filename in double quotation marks ("") if it contains embedded spaces or commas. You can append command line parameters to the filename on the Run script command (within the double quotes, separated by a space). filename may be a variable containing the full filename of an application or batch file.
HIDDEN	run the application with its window hidden.
MINIMIZED	run the application with its window minimized.
MAXIMIZED	run the application with its window maximized.

filename may be a variable.

The **run** command runs the specified application or batch file. filename may specify a .com, .exe, .bat or .pif file.

Examples

```
run("c:\batch\tasks job.bat")
run(c:\skey.exe,MINIMIZED)
run($PROGRAM,HIDDEN)
run("C:\Program Files\SDI\TN3270 Plus\tn3270.exe mainframe")
```

See Also:

Script Command Summary
Script Variables

Using the Scripting Language

6.3.1.47 RunDirectory

```
RunDirectory(directory)
```

Where:

directory	is a directory. Enclose the directory in double quotation marks ("") if it contains embedded spaces or commas. directory may be variable containing a directory name.

The **RunDirectory** command specifies the working directory for the application started by the **run** command. This is useful for applications that use a working directory for their files.

Examples

```
RunDirectory("c:\program files\data")
RunDirectory(c:\test\data)
RunDirectory($DIRECTORY)
```

The following script starts the program test.exe with a working directory of c:\program\data.

Sample Script

```
' This script starts the program test.exe
' with a working directory of c:\program\data.
'
RunDirectory(c:\program\data)
Run(c:\program\test.exe)
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.48 Session

```
session(session_number | session_name)
```

Where:

session_number	is the session number. Session numbers 1 to 99 are valid. session_number may be a variable containing a valid session number.
session_name	is the session name. session_name may be a variable containing a valid session name. If you are using a session name, the session must be running when the command is executed or an error occurs.

The **session** command switches sessions while a script is processing. A script continues processing in the same session until a **session** command switches the script to another session.

Examples

```
session(1)
session(2)
session("mainframe2")
session($SESSION)
```

See Also:

Script Command Summary

Script Variables

Using the Scripting Language

6.3.1.49 SetEnv

```
SetEnv(environment_variable,value)
```

Where:

environment_variable	is a local environment variable name. The environment_variable may not contain embedded spaces. If the environment_variable does not already exist, it is created.
value	the value assigned to environment_variable.

environment_variable and value may be TN3270 Plus variables.

The **SetEnv** command sets the value of a local environment variable. The local environment variable value is then accessible by other TN3270 Plus scripts. This makes it easy to pass information from one script to another. To access the value of a local environment variable, enclose the variable name in percent signs. Local environment variables persist until TN3270 Plus is closed.

Examples

```
SetEnv(PartNumber,78654)
SetEnv(Description,"Widget Maker")
SetEnv(RetailPrice,28.95)
MsgBox(%PartNumber%)
```

Sample Scripts

The first sample script assigns a value to a local environment variable.
The second sample script displays the value of the local environment variable in a message box.

```
' SetLocalEnvironmentVariable.txt
' Example that sets a local environment variable
SetEnv(PartNumber,78654)
exit
```

```
' DisplayLocalEnvironmentVariable.txt
' Example that displays a local environment variable
' in a message box
MsgBox(%PartNumber%)
exit
```

See Also:

Script Variables

SetUserEnv

Using the Scripting Language

6.3.1.50 SetPrt

```
SetPrt({"printer"|"Application Default"|"RESTORE"}[,{P|L}][,Tray(tray_name)
]
```

Where:

printer	is the name of the printer.
Application Default	specifies application default printer. (Host, Print Setup...).
RESTORE	restore the original session printer setting.
P	sets the printer orientation to Portrait. Portrait is the default.
L	sets the printer orientation to Landscape.

tray_name is the name of the paper source as found in the "Host,Print Setup..." dialog. The tray name must be enclosed in double quotes if it contains spaces.

The **SetPrt** command changes the session printer.

Examples

```
SetPrt("HP Printer 1",L,Tray("Tray 2"))
SetPrt("Application Default",P)
SetPrt("RESTORE")
```

Sample Script

The following script sets the printer to "HP Printer 1" with landscape orientation, prints the terminal screen and restores the session back to its default printer.

```
' This script sets the printer to "HP Printer 1" with landscape
                orientation,
' prints the terminal screen and restores the session back to its
                default printer.
'
SetPrt("HP Printer 1",L)
key(HostPrintScreen)
SetPrt("RESTORE")
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.51 SetUserEnv

```
SetUserEnv(environment_variable,value )
```

Where:

environment_variable is a user environment variable name. The environment_variable may not contain embedded spaces. If the environment_variable does not already exist, it is created.
value the value assigned to environment_variable.

environment_variable and value may be TN3270 Plus variables.

The **SetUserEnv** command sets the value of a user environmental variable. This user environment variable is then accessible by external programs like PowerShell or VBScript. This makes it easy to pass information from a TN3270 Plus script to an external program. The user environment variables are visible in the "HKEY_CURRENT_USER\Environment" registry key.

Examples

```
SetUserEnv(ProductNumber,3270)
SetUserEnv(Description,"Terminal Emulator")
MsgBox(%ProductNumber%)
```

Sample Scripts

The following TN3270 Plus sample script assigns a value to a user environment variable.

```
' SetUserEnvironmentVariable.txt
' Example that sets a user environment variable
SetUserEnv(ProductNumber,3270)
```

```
exit
```

The following VBscript accesses the user environment variable created in the TN3270 Plus script above.

```
'DisplayUserEnvironmentVariable.vbs
' Windows VBScript to display a user environment variable

Dim objWSH
Dim objUserVariables

Set objWSH = CreateObject( "WScript.Shell" )
Set objUserVariables = objWSH.Environment("USER")
MsgBox(objUserVariables("ProductNumber"))
```

The following PowerShell script accesses the user environment variable created in the TN3270 Plus script above.

```
# DisplayUserEnvironmentVariable.ps1
# Windows PowerShell script to display a user environment variable

[environment]::GetEnvironmentVariable("ProductNumber", "User")
```

See Also:

Script Variables
SetEnv
Using the Scripting Language

6.3.1.52 SSLConnect

```
SSLConnect({TLSv1 | TLSv1.1 | TLSv1.2 | $variable})
```

Where:

TLSv1	is TLS version 1.0.
TLSv1.1	is TLS version 1.1.
TLSv1.2	is TLS version 1.2.
$variable	is a variable. The text defined for the variable is substituted for the variable name and used as the command parameter.

Use the SSLConnect command to begin a TLS connection when the host computer resides behind a proxy server or firewall.

If the host is behind a proxy server and a TLS connection is required then the connection must be made using a script instead of specifying TLS in the Host pane of the **Session Setup** dialog box. The reason for this is that if the TLS connection is made in the normal manner (using the radio buttons in the Host pane of the **Session Setup** dialog box) then the TLS handshake will take place with the proxy server instead of the host. Insert the **SSLConnect** command at the appropriate place in the script to cause the TLS connection to be made.

Examples

```
SSLConnect(TLSv1)
SSLConnect(TLSv1.1)
SSLConnect(TLSv1.2)
```

Sample Script

```
' The following is a sample using the SSLConnect command
'
' Proxy Server login
WaitFor(username)
type(john)
key(enter)
```

```
WaitFor(password)
type(smith)
key(enter)

' Login complete, connect to host
waitfor("Login complete")
type(connect 127.0.0.1)
key(enter)

' Initiate TLS handshaking
SSLConnect(TLSv1)
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.53 TextBox

```
TextBox("prompt"["caption"] [,BackButton(label)] [,NOBUTTONS]
      [,WindowPos(CENTER)])
      [,WindowPos(CENTRE)]
      [,WindowPos(DEFAULT)]
      [,WindowPos(CURSOR)]
      [,WindowPos(top,left[,width,height][,PIXELS])]
      [,WindowPos(row,col[,width,height],ROWCOL)]
      [,WindowPos(row,col[,width,height],CURSOR)])
```

Where:

prompt	is the text to display in the text dialog box. This parameter may be a variable containing the message text.
caption	is the caption that appears in the title bar of the message box. If omitted, the caption will be "TN3270 Plus". This parameter may be a variable containing the message box caption.
BackButton(label)	If specified, the **Cancel** button in the TextBox dialog is replaced by the **Back** button. If the Back button is clicked, the script will continue at the specified label.
NOBUTTONS	displays the **Text Box** without any buttons. Click the red X in the top right corner to close the dialog box and continue the script.
WindowPos(CENTER)	centers the **Text Box** dialog box over the TN3270 Plus terminal window.
WindowPos(CENTRE)	centers the **Text Box** dialog box over the TN3270 Plus terminal window.
WindowPos(DEFAULT)	centers the **Text Box** dialog box over the TN3270 Plus terminal window. This is the default if the WindowPos parameter is not specified.
WindowPos(CURSOR)	positions the top left corner of the **Text Box** dialog box at the cursor location.
WindowPos(top,left[,width, height][,PIXELS])	positions the top left corner of the **Text Box** dialog box at the specified pixel offset from the top left corner of the TN3270 Plus window. top and left are the offset in pixels. The offset may be negative to move the dialog box outside the TN3270 Plus window. width and height define the size of the dialog box in pixels.
WindowPos(row,col[,width, height],ROWCOL)	positions the top left corner of the **Text Box** dialog box at the specified row and column in the TN3270 Plus terminal window. width and height define the size of the dialog box in terminal rows and columns.

WindowPos(row,col[,width, height],CURSOR)	positions the top left corner of the **Text Box** dialog box at the specified row and column offset from the cursor location in the TN3270 Plus terminal window. width and height define the size of the dialog box in terminal rows and columns.

prompt, caption, top, left, row, col, width and height may be variables.

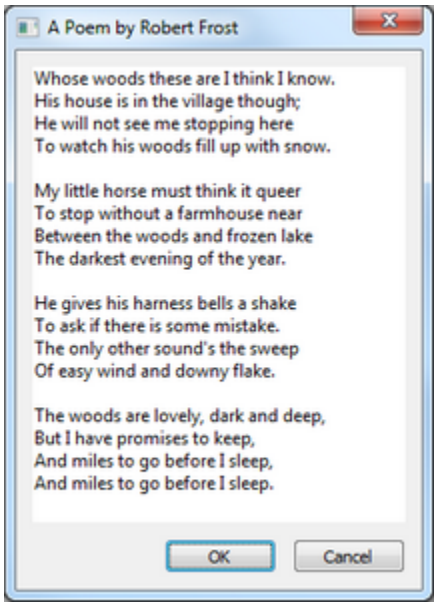
The **TextBox** command pauses the script and displays a dialog box that contains a message for the user. The **Text Box** dialog box displays the prompt and **OK** and **Cancel** buttons. Click the OK button to continue script processing. Click the Cancel button to cancel the script. If the "NOBUTTONS" option is used, the window is closed by clicking the red X in the top right corner and the script continues (it cannot be canceled).

Examples

```
TextBox("Process complete.")
TextBox("Process complete.","File Transfer")
TextBox("Process complete.",NOBUTTONS)
TextBox("Process complete. Click Back button to repeat the process.","Compute",BACKBU
TextBox("Process complete.",NOBUTTONS,WindowPos(10,10))
```

Sample Script

```
$Prompt = "Whose woods these are I think I know." & chr(10) & _
         "His house is in the village though;" & chr(10) & _
         "He will not see me stopping here" & chr(10) & _
         "To watch his woods fill up with snow." & chr(10) & chr(10) & _
         "My little horse must think it queer" & chr(10) & _
         "To stop without a farmhouse near" & chr(10) & _
         "Between the woods and frozen lake" & chr(10) & _
         "The darkest evening of the year." & chr(10) & chr(10) & _
         "He gives his harness bells a shake" & chr(10) & _
         "To ask if there is some mistake." & chr(10) & _
         "The only other sound's the sweep" & chr(10) & _
         "Of easy wind and downy flake." & chr(10) & chr(10) & _
         "The woods are lovely, dark and deep," & chr(10) & _
         "But I have promises to keep," & chr(10) & _
         "And miles to go before I sleep," & chr(10) & _
         "And miles to go before I sleep."
TextBox($Prompt,"A Poem by Robert Frost")
```



See Also:

MsgBox
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.54 Trim

```
$return = trim(string[,trimCharacter[,LEFT | RIGHT]])
```

Where:

$return	the trimmed string.
string	text to be trimmed.
trimCharacter	the leading and/or trailing character(s) to trim from string. If this parameter contains multiple characters all of the occurrences of these character will be trimmed. If trimCharacter is not specified, then it defaults to a space character. This parameter may be a variable containing the trim character or characters. Enclose trimCharacter in double quotation marks ("") if it contains spaces or commas.
LEFT	trim leading characters from string. If neither LEFT or RIGHT is specified, then both leading and trailing characters are trimmed.
RIGHT	trim trailing characters from string.

The **trim** command trims leading and/or trailing characters from a variable.

Example

```
$return = trim($text)
$return = trim($text,0,left)
$return = trim($text," ", right)
$return = trim($text,"$ ")
```

Sample Script

```
' trim.txt
' Trim trailing zeros.
```

```
$text = "985.2000000"
$return = trim($text,"0",Right)
MsgBox($return)
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.55 Type

```
type("string1" [& "string2" [& "string3"...]])
```

Where:

stringn	is a string of characters to be typed into the session. This field is case sensitive. Enclose stringn in double quotation marks ("") if it contains embedded spaces or commas. If a double quote (") needs to be included in stringn enter two double quotes (""). The two double quotes will be replaced by a single double quote when the string is typed. Any of the stringn parameters may be variables. If stringn is a variable the text associated with the variable is substituted and typed into the session at the current cursor location.

The **type** command enters the string into a session at the current cursor location. If multiple stringn parameters are specified they are concatenated and typed into the session.

Examples

```
type("query names")
type($NAME)
type("This is how to include ""double quotes"" in the text")
type($CLIPBOARD)
type($FIRSTNAME & " " & $LASTNAME)
```

Sample Script

```
' This script enters the "query names" command into a session.
'
type("query names")
key(enter)
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.56 UCase

```
$return = UCase(string)
```

Where:

$return	receives upper case string.
string	a string of characters.

The **UCase** command converts the lower case characters in the string to upper case characters. Upper case

characters and numeric characters are not changed.

Examples

```
$return = ucase("This Is A Test")
```

See Also:

LCase
Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.57 Wait

```
wait(milliseconds)
```

Where:

milliseconds	is the number of milliseconds script processing should wait before continuing with the next command. milliseconds may be a variable containing the number of milliseconds.

The **wait** command pauses script processing for the specified number of milliseconds. If you wish to wait for a host response before continuing script processing, the **wait** command may not be the best choice. Since response times are difficult to predict and inconsistent, the wait must be long enough for the longest possible response. This is not very efficient. It may be better to use WaitFor or AskFor to pause the script.

Examples

```
wait(1000)
wait(2000)
wait($TIME)
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.58 WaitFor

```
WaitFor(string1[,label1:] [,string2,label2:...,string10,label10:]
        [,minimized | hidden][,WindowPos(CENTER)])
                             [,WindowPos(CENTRE)]
                             [,WindowPos(DEFAULT)]
                             [,WindowPos(CURSOR)]
                             [,WindowPos(top,left[,PIXELS])]
                             [,WindowPos(row,col,ROWCOL)]
                             [,WindowPos(row,col,CURSOR)]
```

Where:

stringn	is a string of characters. stringn is case sensitive. Enclose stringn in double quotation marks ("") if it contains embedded spaces or commas.
labeln	is the name of the label in the script. Labeln: is case sensitive.
minimized	minimize the **WaitFor** dialog box.
hidden	hides the **WaitFor** dialog box, so the user will not see it displayed.

WindowPos(CENTER)	centers the **WaitFor** dialog box over the TN3270 Plus terminal window.
WindowPos(CENTRE)	centers the **WaitFor** dialog box over the TN3270 Plus terminal window.
WindowPos(DEFAULT)	centers the **WaitFor** dialog box over the TN3270 Plus terminal window.
WindowPos(CURSOR)	positions the top left corner of the **Waitfor** dialog box at the cursor location.
WindowPos(top,left[,PIXELS])	positions the top left corner of the **WaitFor** dialog box at the specified pixel offset from the top left corner of the TN3270 Plus window. top and left are the offset in pixels. The offset may be negative to move the dialog box outside the TN3270 Plus window.
WindowPos(row,col ,ROWCOL)	positions the top left corner of the **WaitFor** dialog box at the specified row and column in the TN3270 Plus terminal window.
WindowPos(row,col ,CURSOR)	positions the top left corner of the **WaitFor** dialog box at the specified row and column offset from the cursor location in the TN3270 Plus terminal window.

stringn, labeln, top, left, row and col may be variables.

The **WaitFor** command accepts from 1 to 10 string,label: pairs.

WaitFor with a single parameter

Examples

```
WaitFor(USERID,minimized)
WaitFor( ===>,hidden)
WaitFor($NAME)
WaitFor("Test text",WindowPos(10,10,ROWCOL))
WaitFor("")
```

When the **WaitFor** command is used with a single string parameter (no label), the **WaitFor** command repeatedly scans the session screen buffer for the specified string. When the string is found, script processing continues with the next command. The **WaitFor** command displays a dialog box containing the scan string. If the scan cannot locate the string, the user can click the **Cancel Script** button in the dialog box to cancel the script or the **Cancel Scan** button to cancel the scan and continue with the next script command.

If the Waitfor command is specified with a null string parameter:

```
Waitfor("")
```

the script waits until the next time the host updates the screen and then continues with the next command.

The purpose of this Waitfor command is to wait for a screen response from the host and then continue script processing. Use this command to prevent a script from issuing commands before the host is ready to receive them.

WaitFor with multiple parameters

Examples:

```
WaitFor(RECONNECTED,LABEL1:,Ready,LABEL2:)
WaitFor(abcd,LA:,efgh,LE:,ijkl,LI:,mnop,LM:,qrst,LQ:,uvwx,LU:,yz,LY:)
```

When the **WaitFor** command is used with multiple parameters, the **WaitFor** command repeatedly scans the session screen buffer looking for the string(s). When it finds one of the strings, it branches to the associated label. The purpose of this command is to wait for a screen response from the host and then take the appropriate action based upon what the host returns.

Warning:

The **WaitFor** command scans the entire screen buffer each time it is updated. If one of the strings in the **WaitFor** parameters is already on the screen when the **WaitFor** command is issued, the search is satisfied immediately. This defeats the purpose of waiting for a host response. Make sure any string you specify will not be on the screen when the **WaitFor** command is issued.

Sample Script

```
'The WaitFor command scans the session screen buffer for
'the strings RECONNECTED" or "Ready". If the WaitFor command
'finds "RECONNECTED", it 'branches to "LABEL1:." If the WaitFor
'command finds "Ready", it branches to "LABEL2:."

WaitFor(RECONNECTED,LABEL1:,Ready,LABEL2:)
LABEL1:
type(begin)
key(enter)
goto EXIT:
LABEL2:
type(test)
key(enter)
EXIT:
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.1.59 Weekday

```
$return = Weekday(date[,firstdayofweek][,date_format])
```

Where:

$return	A number from 1 to 7 representing the day of the week.
date	The input date.
firstdayofweek	A number between 1 and 7 that specifies the day to use as the first day of the week. 1 for Sunday (default) 2 for Monday 3 for Tuesday 4 for Wednesday 5 for Thursday 6 for Friday 7 for Saturday
date_format	can be one of the following: DMY (date is in DD/MM/YYYY format.) MDY (date is in MM/DD/YYYY format.) YMD (date is in YYYY/MM/DD or YYYY-MM-DD format.) date_format may be omitted, in which case the date format is taken from the "Option DateFormat" command. If the "Option DateFormat" command is also omitted, then the date format is assumed to be the system short date format.

date, firstdayofweek and/or date_format may be variables.

The **Weekday** command returns a number between 1 and 7 for the day of the week of the input date.

Examples

```
$weekday = weekday($DATE)              'Returns today's weekday number in
                                       system short date format.
$weekday = Weekday("2015-01-01",       'Returns 05
YMD)
$weekday = Weekday("2015-01-02",       'Returns 06
   ymd)
$weekday = Weekday("2015-01-01",       'Returns 04
   ymd, 2)
```

Sample Script

```
 ' Weekday - Display the day of the week.
MsgBox("Today is = " & WeekdayName($DATE)
exit
```

See Also:

DateAdd
Day
Month
MonthName
WeekdayName
Year

6.3.1.60 WeekdayName

```
$return = WeekdayName(weekday[,abbreviate[,firstdayofweek]])
$return = WeekdayName(date[,abbreviate][,date_format])
```

Where:

$return	The weekday name of the specified day the week. (For example, Sunday)
weekday	The number of the the day of the week. See the Weekday script command.
date	The input date,
abbreviate	Specify "true" to return an abbreviated weekday name. Specify "false" to return the full weekday name. ("false" is the default)
firstdayofweek	A number from 0 to 7 that specifies the day to use as the first day of the week. 0 for the first day of the week as defined by the systems settings. 1 for Sunday (default) 2 for Monday 3 for Tuesday 4 for Wednesday 5 for Thursday 6 for Friday 7 for Saturday
date_format	can be one of the following: DMY (date is in DD/MM/YYYY format.) MDY (date is in MM/DD/YYYY format.) YMD (date is in YYYY/MM/DD or YYYY-MM-DD format.) date_format may be omitted, in which case the date format is taken from the "Option DateFormat" command. If the "Option DateFormat" command is also omitted, then the date format is assumed to be the system short date format.

weekday, date, firstdayofweek and/or date_format may be variables.

The **WeekdayName** command returns the weekday name for the specified day of the week.

Examples

```
$weekday = WeekdayName($DATE)            'Returns today's weekday name
                                           in system short date format.
$weekday = WeekdayName(1)                'Returns Sunday
$weekday = WeekdayName(1, TRUE)          'Returns Sun
$weekday = WeekdayName(1, false, 2)      'Returns Monday
$weekday =                               'Returns Thursday
   WeekdayName("2015-01-01",false,ymd)
$weekday =                               'Returns Fri
   WeekdayName("2015-01-02",true,YMD)
```

Sample Script

```
' Weekday - Display the day of the week.
MsgBox("Today is = " & WeekdayName($DATE))
exit
```

See Also:

DateAdd
Day
Month
MonthName
Weekday
Year

6.3.1.61 Year

```
$return = Year(date[,date_format])
```

Where:

$return	The four digit year.
date	The input date.
date_format	can be one of the following: DMY (date is in DD/MM/YYYY format.) MDY (date is in MM/DD/YYYY format.) YMD (date is in YYYY/MM/DD or YYYY-MM-DD format.) date_format may be omitted, in which case the date format is taken from the "Option DateFormat" command. If the "Option DateFormat" command is also omitted, then the date format is assumed to be the system short date format.

date and date_format may be a variables.

The **Year** command returns the four digit year of the input date.

Examples

```
$Year = Year($DATE)              'Returns the current year in system
                                 short date format
$Year = Year("2015-01-01",       'Returns 2015
YMD)
$Year = Year("01/01/2016",       'Returns 2016
mdy)
```

Sample Script

```
' Display the date.
$Today = $DATE
$DayNumber = Day($Today)
$MonNumber = Month($Today)
$Year = Year($Today)
'Remove leading zero
$DayNumber = Trim($DayNumber,"0",left)
$MonNumber = Trim($MonNumber,"0",left)
MsgBox("Today is day "  & $DayNumber _
         & " of month " & $MonNumber _
         & " of year "  & $Year)
exit
```

See Also:

DateAdd
Day
Month
MonthName
Weekday
WeekdayName

6.3.2 Script Operators (Release 3.6 and above)

Concatenation Operator

Operator	Description	Example	Result
&	string concatenation	"ab" & "c"	"abc"

Math Operators

Operator	Description	Example	Result
+	Addition	2 + 3	5
-	Subtraction	3 - 2	1
*	Multiplication	2 * 3	6
/	Divide	4 / 2	2
%	Modulus	3 % 2	1

Comparison Operators

Operator	Description	Example	Result
=, EQ	Equal to	2 = 2	true
>, GT	Greater than	3 GT 2	true
<, LT	Less than	2 < 3	true

>= , GE	Greater than or equal to	3 GE 2	true
<=, LE	Less than or equal to	2 <= 2	true
<>, NE	Not equal to	1 NE 2	true
~, LIKE	Matches pattern containing wildcards	123 ~ ###	true

Wildcards for the LIKE Operator

Wildcard Characters	Matches in the comparison string
?	Any single character
*	Zero or more characters
#	Any single numeric digit
[charlist]	Any single character in charlist
[!charlist]	Any single character not in charlist

charlist is a group of one or more characters enclosed in brackets ([]) and can be used to match any single character in a string. It may include any of the following characters:

- Almost any character in the ANSI character set including digits
- The special characters opening bracket ([), question mark (?), number sign (#) and asterisk (*) can be used for matching only if enclosed in brackets. The closing bracket (]) can't be used within a charlist to match itself, but it can be used outside the group as an individual character. A hyphen (-) can be used at the beginning or end of charlist to match itself. In any other position it indicates a range of characters.
- A range of characters can be specified by placing a hyphen between lower and upper bounds. For example, [a-z], [A-Z] or [0-9]. Multiple ranges can be specified without any delimiter. For example [a-zA-Z0-9] matches any alphanumeric character. When a range of characters is specified they must appear in ascending sort order. [A-Z] is valid, but[Z-A] is not.

6.3.3 Script Variables (Release 3.6 and above)

The script language has four types of variables: built-in variables, Windows environment variables, local variables and global variables.

Built-in Variables

Built-in variables are predefined. All built-in variables start with a dollar sign ($) and are all upper case. The following built-in variables are available.

`$ARG0`	full path of the script file from the command line.
`$ARG1, ARG2, ...`	arguments from the command line.
`$ARGC`	count of arguments on the command line.
`$CLIPBOARD`	replaced by any text on the Windows clipboard.
`$COLS`	replaced by the number of screen columns.
`$COMPUTERNAME`	computer name.

$CRLF	replaced by a carriage return and line feed.
$CURSOR	replaced by the cursor position on the screen relative to 1 (row 1 column 1).
$DATE	date in the local date format.
$DATEDMY	date in DD/MM/YYYY format.
$DATEMDY	date in MM/DD/YYYY format.
$DATEYMD	date in YYYY-MM-DD format.
$FILEERROR	the return code of the last script file I/O operation.
$LONGDATE	date in local long date format.
$OIA	replaced by the text in the Operator Information Area line on the terminal screen
$PARENTSCRIPT	name of the calling script or "<none>".
$ROWS	replaced by the number of screen rows.
$SCREEN[(start[,length])]	replaced by the text at the specified location on the terminal screen. where: start — is the starting position on the screen. Specify 1 to indicate row 1 column 1. length — is the number of characters to include. start and/or length may be a variable. The ((start[,length])) parameter is optional. If you specify "$SCREEN" with no parameter, it is replaced by the text contents of the entire screen. If the (start[,length]) parameter is invalid, no substitution will take place and the variable will be treated as a literal.
$SCREEN[(top,left,bottom,right[,RECT])]	replaced by the text at the specified location on the terminal screen. where: top — is the first row of the selection rectangle. left — is the left column of the selection rectangle. bottom — is the last row of the selection rectangle. right — is the right column of the selection rectangle. RECT — is optional and specifies that the selection area is a rectangle, not a string. top, left, bottom, right and/or RECT may be a variable. The (top,left,bottom,right) parameter is optional. If you specify "$SCREEN" with no parameter, the variable is replaced by the text contents of the entire screen. If the (top,left,bottom,right) parameter is invalid, no substitution will take place and the variable will be treated as a literal.

$SCREEN[(top,left,length)]	replaced by the text at the specified location on the terminal screen. where: top is the first row of the selection rectangle. left is the left column of the selection rectangle. length is the number of characters to include. top, left and/or length may be a variable. The (top,left,length) parameter is optional. If you specify "$SCREEN" with no parameter, the variable is replaced by the text contents of the entire screen. If the (top,left,length) parameter is invalid, no substitution will take place and the variable will be treated as a literal.
$SCRIPTFOLDER	the full path of the folder containing the currently active script.
$SCRIPTNAME	name of currently active script.
$SESSIONNAME	name of session the script is running in or "<none>".
$TIME	time in local time format.
$USERNAME	current logged on user name.
$VERSION	The TN3270 Plus version number as an integer. For example release 3.6.0 = 3600 and release 3.6.1 = 3601.

Sample Script

```
' ----------------------------------------
' Display Script Variables
' ----------------------------------------

MsgBox(_
  "ARGC = " & chr(9) & $ARGC & $CRLF &  _
  "$COLS = " & chr(9)& $COLS & $CRLF & _
  "$COMPUTERNAME = " & $COMPUTERNAME & $CRLF &  _
  "$CURSOR = " & chr(9)& $CURSOR & $CRLF & _
  "$DATE = " & chr(9)& $DATE & $CRLF & _
  "$FILEERROR = " & chr(9)& $FILEERROR & $CRLF & _
  "$LONGDATE = " & chr(9)& $LONGDATE & $CRLF & _
  "$OIA = " & chr(9)& $OIA & $CRLF & _
  "PARENTSCRIPT = " & chr(9)& $PARENTSCRIPT & $CRLF & _
  "ROWS = " & chr(9)& $ROWS & $CRLF & _
  "SCREEN(1,20) = " & chr(9)& $SCREEN(1,20) & $CRLF & _
  "$SCRIPTFOLDER = " & chr(9)& $SCRIPTFOLDER & $CRLF & _
  "$SCRIPTNAME = " & chr(9)& $SCRIPTNAME & $CRLF & _
  "$SESSIONNAME = " & chr(9)& $SESSIONNAME & $CRLF & _
  "$TIME = " & chr(9)& $TIME & $CRLF & _
  "$USERNAME = " & chr(9)& $USERNAME & $CRLF & _
  "$VERSION = " & chr(9)& $VERSION & $CRLF &_
)

exit
```

Windows Environment Variables

Windows environment variables can be used in script strings. Windows environment variables are enclosed in % signs and may be in upper or lower case. For example:

```
type(%COMPUTERNAME%)
```

Windows User Environment Variables

Windows user environment variables can be used in script strings. Windows user environment variables are enclosed in % signs and may be in upper or lower case. Windows user environment variables can be created with the TN3270 Plus script SetUserEnv command. Windows User Environment Variables are accessible from the programs external to TN3270 Plus like Windows PowerShell or VBScript.

Local Environment Variables

Local Environment variables can be used in script strings. Local environment variables are enclosed in % signs and may be in upper or lower case. Local environment variables can be created with the TN3270 Plus script SetEnv command. Use Local Environment Variables to pass data from one TN3270 Plus script to another.

Local Variables

Local variable names start with a dollar sign ($) and may be mixed case. Local variables are created dynamically by an assignment statement. For example:

```
$Name = AskFor("Enter your userid and click OK")
$CursorRow = Convert($Cursor,ROW)
$CursorColumn = Convert($Cursor,COL)
```

Most script commands accept a variable in place of literal data. The text contained in the variable is substituted for the variable when the script command is processed.

Global Variables

Global variable use the same syntax as a local variable. The variable name must start with a dollar sign ($) and may be mixed case. Global variables must be declared using the Global script command. The data in global variables is available within scripts called using the include script command. Local variables are not available to scripts called using the include command.

Variable Substring

You can extract a substring from a local variable, global variable or the built-in variable using the following format:

```
$variable[(start[,length])]
```

where:

start is the starting position within the variable.

length is the number of characters to include. length defaults to 1 if it is not specified.

start and/or length may be a variable.
For example, the following script types **Joh** on the screen.

```
$Test = "John Doe"
type($Test(1,3))
exit
```

Concatenated Variable Names

You can concatenate variable names. For example, $$SCREEN(1,4)$SCREEN(10,3) would result in a variable name $ followed by the contents of screen location 1-4 followed by the contents of screen location 10-12. Thus, if screen location 1-4 contained "WHIZ" and screen location 10-12 contained "KID" the result would be a variable

name of "$WHIZKID".

See Also:

Script Command Summary
Using the Scripting Language

6.3.4 Script Command Line Arguments

It is possible to pass arguments to a TN3270 Plus script. Arguments are passed on the command line following the full path to the script file. Arguments are separated by spaces. If any of the arguments contain spaces they must be enclosed in double quotes.

Within the script the following built-in variables hold the command line arguments.

$ARGC count of arguments on the command line.

$ARG0 full path of the script file from the command line.

$ARG1, $ARG2... arguments from the command line.

The following script displays the command line arguments passed by the caller.

```
' DisplayArguments.txt
' Sample script to display arguments passed on the script command line.
MsgBox("Argument Count =  " & $ARGC & $CRLF)
MsgBox("Script Path  =  " & $ARG0 & $CRLF)
$Count = 1
Loop:
If $Count < $ARGC then
   MsgBox("$ARG" & $Count & " = " & $ARG$Count)
   $Count = $Count + 1
else
   Exit
End If
GoTo Loop:
Exit
```

You cannot pass arguments via the "Host, Run Script" command but you can specify them on the startup script file (Setup, Host, Script File).

Calling a TN3270 Plus Script from a Windows VB Script

Here is an example of calling a TN3270 Plus script with command line arguments from a VB Script:

```
' Call a TN3270 Plus Script from a Windows VB Script
' Language: VBScript using the Windows Script Host
' When the objShell.Run command parameter contains double quotes within the parameter,
' substitute each double quote in the parameter with two double quotes.
dim objShell
set objShell = WScript.CreateObject("WScript.Shell")
objShell.Run """c:\Program Files (x86)\SDI\TN3270 Plus\tnrun.exe""" _
      & "mainframe script c:\scripts\DisplayArguments.txt ONE TWO THREE"
```

6.3.5 Script Commands (Release 3.5 and below)

6.3.5.1 Using the Scripting Language (Release 3.5 and below)

The TN3270 Plus scripting language is designed to automate common tasks and keystroke sequences. Each day the same keystrokes are entered over and over again. By creating a script, common sequences of keystrokes can be stored in a file. When the script file is run, the keystrokes are entered automatically.

Here are some common uses for scripts:

- Connecting terminal emulation sessions.
- Logging on terminal emulation users.
- Transferring files from the PC to the Host or from the Host to the PC.
- Performing a series of commonly issued commands.

A script file is a text file (.txt) that contains script commands. A script simulates the activities of a user at the keyboard. The script file can be created using your favorite word processor or text editor. Any application than can create a standard text (.txt) file will work.

Script Language Syntax Definition

The syntax descriptions of the script commands use the following notational conventions:

1. **Magenta** colored words are replaced by user input.
2. **Blue** colored symbols are part of the syntax definition and are not included in the command.
3. The following symbols are part of the command and should be entered exactly as they appear in the command format:
   , comma
   : colon
   " double quotation marks
4. Square brackets **[ ]** indicate an optional parameter and are not included in the command.
5. Braces **{ }** are used with the vertical bar **|** to indicate choices between two or more mutually exclusive items and are not included in the command.
6. Words starting with a dollar sign **$** are variables. Variable names are case sensitive.
7. Words starting in column one and ending with a colon **:** are labels. Label names are case sensitive.
8. An ellipsis **...** indicates that the same pattern continues.
9. Lines starting with an asterisk ***** or white space in column one are treated as comments.

Each line in a script file contains a command in the following format:

[label] command [parameter1[,parameter2,parameter3,...parametern]] comment

The command indicates the function to perform and the parameter(s) provide the data needed for the function. Anything following white space after the last parameter on each line is a comment. Parameters containing embedded blanks or commas must be enclosed in double quotation marks. Parameters may be split across multiple lines. A comma followed by white space indicates more parameters on the next line. For example,

```
waitfor   "Reconnected",LABEL1:,   this is a comment
          "Ready", LABEL2:         this is a comment
```

The command is not case sensitive, so it may appear in any combination of upper and lower case. Any blanks or tabs between the command and the first parameter are ignored.

Comment lines can be added to a script file by placing an asterisk (*) in column 1. For example:

```
* ------------------------------------------------------------------
* These three lines are comments
* ------------------------------------------------------------------
```

The Script Commands section contains a complete list of script commands. The syntax of each command is presented along with examples for using the command.

Creating Your First Script

The following steps will lead you through the creation of your first script:

1. Create a new text file in the TN3270 Plus directory using you favorite word processor or text editor.
2. Type the following lines:

```
* This is a comment. -- My first script file.
type   "this is a test"
key   enter
exit
```

3. Save this script as a text file named "test.txt" in the TN3270 Plus directory. Your first script is now complete.

Running Your First Script

The following steps will lead you through running your first script:

1. Connect a TN3270 Plus session.
2. Logon to your userid.
3. Open the **Host** menu and select **Run Script...** and click on the **Browse…** button.
4. In the **Select Script File** dialog box, click on "test.txt" and then click on the **Open** button.
5. "this is a test" is typed on your screen and then "entered" (pressing the enter key is simulated by the script). The result of this script is the same as if you typed "this is a test" on the keyboard and pressed the enter key.

For an example of a logon script, see the Sample Logon Script.

6.3.5.2 Script Command Summary (Release 3.5 and below)

The following is a list of script commands. For an introduction to writing a script and the script command syntax see Using the Scripting Language.

Command	Description
AskFor	display a dialog box requesting user input.
CalcVar	do simple arithmetic on a variable.
command	issue a command in the Windows command environment.
Connect	connect to a host using a named session.
CursorTo	move the cursor.
DDE	perform a complete DDE conversation in a single command.
DDEExecute	send a command to a DDE server application.
DDEInitiate	begin a DDE conversation with a DDE server application.
DDEPoke	send text to a DDE server application.
DDERequest	request text from a DDE server application.
DDETerminate	close a DDE channel.
EditSelect	select data for cut and copy.
Exit	exit the script.
FileSpec	specify the file to be used in a HostSave, HostPrintScreen, HostEnableLogging or EditSaveClipboard command.
FileTransfer	initiate a file transfer.
Find	find a substring inside a string.
GetField	extract a field from a field-separated string.
GetString	read a string from a file into a variable.

global	define a global variable.
goto	branch to a label in the script file.
if	branch to a label in the script file based in the results of a comparison.
include	call another script file.
key	simulate a function key.
MsgBox	display a dialog box.
option	set script options.
PutString	write a string to a file.
replace	replace a character or substring in a string.
run	run an application.
RunDirectory	specify the working directory for a program started by the "run" command.
session	switch to another active session.
SetPrt	change the session printer.
SetVar	initialize a variable.
SSLConnect	begin an SSL connection.
trim	trim leading and/or trailing characters from a variable.
type	"type" characters into the session terminal.
wait	wait for the specified number of milliseconds.
WaitFor	wait until the specified text string appears in the host session (WaitFor USERID). Or, wait until one of several specified text strings appears in the host session. When one of the text strings is found, branch to the specified label.

See Also:

Script Variables
Using the Scripting Language

6.3.5.3 AskFor

```
AskFor  "message text"[,password][,$variable][,"default data"][,WindowPos(top,left)]
```

Where:

message text	is text telling the user what to enter into the AskFor dialog box.
password	an optional keyword that indicates asterisks (*) are displayed rather than the text the user types into the **AskFor** dialog box.
$variable	an optional variable name. The data the user types into the **AskFor** dialog box is associated with this variable instead of typed on the screen. If the variable already contains data and no default data is specified on the AskFor command; the variable data appears as the default data in the edit box within the AskFor dialog
default data	this text appears in the edit box within the AskFor dialog. The user can accept this data by pressing enter or change it by typing over it.
WindowPos(top,left)	specifies the position of the top left corner of the AskFor dialog box. top and left are the offset in pixels from the top left corner of the TN3270 Plus main window. The offset may be negative to move the dialog box outside the TN3270 Plus window. If this parameter is not specified, the dialog box is displayed in the center of the TN3270 Plus main window.

top and left may be variables.

The **AskFor** command displays a dialog box requesting information from the user. The **AskFor** dialog box displays the message text and the user accepts the default data or types the requested data into the edit box. When the user clicks the **OK** button the characters from the edit box are typed into the session or placed in the optional $variable. The **AskFor** command is a good way to provide additional information to lead inexperienced

users through a process.

Examples

```
AskFor   "Enter your userid and then click OK",John
AskFor   "Enter your last name and click OK",Doe
AskFor   "Enter your password",password
AskFor   "Enter your name",$NAME,"John Doe",WindowPos(10,10)
```

The following script asks the user for logon information.

```
Askfor   "Enter your userid and click OK"
key      tab
AskFor   "Enter your password and click OK",password
key      enter
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.4 CalcVar

`CalcVar  $variable,operand1,operator,operand2[[,precision][,TRIM]]`

Where:

$variable	is the name of the variable that receives the result.
operand1	is the first operand in the equation. This operand is a decimal number that can contain a single decimal point and one or more commas before the decimal point.
operator	is the operator for the calculation. The following operators are valid: + Add operand1 and operand2. - Subtract operand2 from operand1. * Multiply operand1 by operand2. / Divide operand1 by operand2. The result of a division operation is always rounded down. Use the remainder operator (%) to get the remainder from a division operation. % Divide operand1 by operand2 and return the remainder.
operand2	is the second operand in the equation. This operand is a decimal number that can contain a single decimal point and one or more commas before the decimal point.
precision	is an optional parameter that specifies the number of decimal places to include in the result. The default is zero.
TRIM	is and optional keyword that trims trailing zeros following the decimal point.

operand1, operator, operand2, precision and/or TRIM may be variables.

The **CalcVar** command performs integer or decimal arithmetic and places the result in a variable. If precision is omitted, integer arithmetic is performed.

Examples

```
CalcVar  $Sum,2,+,2
CalcVar  $Difference,8,-,4
CalcVar  $Product,2,*,2
CalcVar  $Quotient,16,/,4
CalcVar  $Quotient,16.5,/,4.5,4,TRIM
CalcVar  $Remainder,3,%,2
CalcVar  $LineNumber,$LineNumber,+,1
CalcVar  $Position,$Position,+,$Columns
```

Sample Script

```
* CalcVar.txt
* Type 1 through 9

SetVar $Count,1

LOOP:
type $Count
CalcVar $Count,$Count,+,1
if $Count,LT,10,LOOP:

exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.5 command

`command` *command*

Where:

command	is a command to be issued in the Windows command environment. Enclose the command in double quotation marks ("") if it contains embedded spaces, tabs or commas. This parameter may be a variable containing the command to be issued.

The **command** script command allows Windows commands to be executed from within a script.

Examples

```
command  "rename test.txt test1.txt"
command  "delete c:\test1.txt"
command  dir
command  $COMMAND
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.6 Connect

`Connect` *"session name"*

Where:

session name	is the name of a session. session names are case sensitive. Enclose the session name in double quotation marks ("") if it contains embedded spaces or commas. This parameter may be a variable containing the session name.

The **Connect** command connects the specified session. Session names can be created and saved in the **Connect to Host** dialog box. The Session Name group box in the **Connect to Host** dialog box contains a drop-down list of the named sessions. Open the **Host** menu and select **Connect...** to display the **Connect to Host** dialog box. The **Connect** command starts the new session in the next available session.

Examples

```
Connect  "Host 1"
Connect  "Host 2"
Connect  Host_2
Connect  $SESSION
```

The following script connects the Host 2 named session.

```
Connect  "Host 2"
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.7 CursorTo

```
CursorTo  position
CursorTo  row,column
```

Where:

position	is the screen position relative to one.
row	is the row number.
column	is the column number.

position, row and/or column may be variables.

The **CursorTo** command moves the cursor to the specified position or the specified row and column. The upper left-hand corner of the screen is position 1 or row 1 column 1.

Examples

```
CursorTo 1
CursorTo 81
CursorTo 1,1
CursorTo 24,80
CursorTo $POSITION
CursorTo $ROW,$COLUMN
```

Restrictions:

CursorTo is not supported for ANSI and VTxxx terminal emulation.

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.8 DDE

```
DDE   {POKE | REQUEST | EXECUTE},service,topic,item{,data | $result}[,
onErrorLabel:]
```

Where:

POKE	sends text to the DDE server application.
REQUEST	requests text from the DDE server application.
EXECUTE	sends a command to the DDE server application.
service	is name of the DDE application for this conversion.

topic	is name of the DDE topic for this conversion.
item	is the name of the DDE item for this conversation.
data	is a string of text to be sent to the DDE server application for a POKE operation.
$result	is a variable that receives the result of a REQUEST operation.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDE command is a shorthand command that performs a complete DDE conversation in a single command. It performs the functions of DDEInitiate, one of DDEPoke, DDERequest or DDEExecute, and DDETerminate. Use this command if you want to perform a single DDE function.

Examples

```
DDE REQUEST,"EXCEL","Sheet2",R1C1,$RESULT
```

Sample Script

```
* The following script starts Microsoft Excel with a test spreadsheet,
* gets the data in row 1 column 1 of the spreadsheet and types it into
* the terminal emulation screen at the current cursor location.
*
Run "C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls"""
Wait 1000
DDE REQUEST,"EXCEL","Sheet2",R1C1,$RESULT
type $RESULT
exit
```

See Also:

DDE
DDEExecute
DDEInitiate
DDEPoke
DDETerminate

6.3.5.9 DDEExecute

```
DDEExecute   $variable,item[,onErrorLabel:]
```

Where:

$variable	is the variable containing the DDE channel number. The DDE channel is established by the DDEInitiate command.
item	is the command sent to the DDE server application.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDEExecute command sends a command to a Dynamic Data Exchange (DDE) server application.

Example

```
DDEExecute   $CHANNEL,"[SAVE()][QUIT()]"
```

Sample Script

```
* The following script starts Microsoft Excel with a test spreadsheet.
* "some data" is then put into row 1 column 1 of the spreadsheet.
* The spreadsheet is saved and Excel is closed.
```

```
* The DDE conversation is terminated.
*
Run "C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls"""
Wait 1000
DDEInitiate  $CHANNEL,"EXCEL","Sheet2"
DDEPoke      $CHANNEL,"R1C1","some data"
DDEExecute   $CHANNEL,"[SAVE()][QUIT()]"
DDETerminate $CHANNEL
exit
```

See Also:

DDE
DDEInitiate
DDEPoke
DDERequest
DDETerminate

6.3.5.10 DDEInitiate

```
DDEInitiate  $variable,service,topic[,onErrorLabel:]
```

Where:

$variable	is the name of the variable that receives the DDE channel number. Subsequent DDE script commands reference the channel number created by DDEInitiate.
service	is the name of the DDE application for this conversation.
topic	is the name of the DDE topic for this conversation.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDEInitiate command begins a Dynamic Data Exchange (DDE) conversation between TN3270 Plus and another application. The DDE conversation allows data to be exchanged between TN3270 Plus and the other application.

Example

```
DDEInitiate $CHANNEL,"EXCEL","Sheet2"
```

Sample Script

```
* The following script starts Microsoft Excel with a test spreadsheet.
* "some data" is then put into row 1 column 1 of the spreadsheet.
* The spreadsheet is saved and Excel is closed.
* The DDE conversation is terminated.
*
Run "C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls"""
Wait 1000
DDEInitiate  $CHANNEL,"EXCEL","Sheet2"
DDEPoke      $CHANNEL,"R1C1","some data"
DDEExecute   $CHANNEL,"[SAVE()][QUIT()]"
DDETerminate $CHANNEL
exit
```

See Also:

DDE

DDEExecute
DDEPoke
DDERequest
DDETerminate

6.3.5.11 DDEPoke

```
DDEPoke   $variable,item,data[,onErrorLabel:]
```

Where:

$variable	is the variable containing the DDE channel number. The DDE channel is established by the DDEInitiate command.
item	is the name of the item to be updated.
data	is a string of text to be sent to the DDE server application.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDEPoke command sends text to a Dynamic Data Exchange (DDE) server application.

Example

```
DDEPoke $CHANNEL,"R1C1","some data"
```

Sample Script

```
* The following script starts Microsoft Excel with a test spreadsheet.
* "some data" is then put into row 1 column 1 of the spreadsheet.
* The spreadsheet is saved and Excel is closed.
* The DDE conversation is terminated.
*
Run "C:\Program Files\Microsoft Office\Office11\excel.exe
      ""D:\Test.xls"""
Wait 1000
DDEInitiate  $CHANNEL,"EXCEL","Sheet2"
DDEPoke      $CHANNEL,"R1C1","some data"
DDEExecute   $CHANNEL,"[SAVE()][QUIT()]"
DDETerminate $CHANNEL
exit
```

See Also:

DDE
DDEExecute
DDEInitiate
DDERequest
DDETerminate

6.3.5.12 DDERequest

```
DDERquest   $variable,item,$result[,onErrorLabel:]
```

Where:

$variable	is the variable containing the DDE channel number. The DDE channel is established by the DDEInitiate command.
item	is the name of the item being requested.
$result	is a variable that receives the result of the request.
onErrorLabel:	is a label that execution transfers to if an error occurs.

The DDERequest command receives text from a Dynamic Data Exchange (DDE) server application.

Example

```
DDERequest $CHANNEL,"R1C1",$RESULT
```

Sample Script

```
* The following script starts Microsoft Excel with a test spreadsheet,
* gets the data in row 1 column 1 of the spreadsheet, terminates excel,
* and types the requested text into the terminal emulation screen at
* the current cursor location.
*
Run "C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls"""
Wait 1000
DDEInitiate  $CHANNEL,"EXCEL","Sheet2"
DDERequest   $CHANNEL,"R1C1",$RESULT
DDEExecute   $CHANNEL,"[QUIT()]"
DDETerminate $CHANNEL
type $RESULT
exit
```

See Also:

DDE
DDEExecute
DDEInitiate
DDEPoke
DDETerminate

6.3.5.13 DDETerminate

```
DDETerminate  $variable
```

Where:

$variable	is the variable containing the DDE channel number. The DDE channel is established by the DDEInitiate command.

The DDETerminate command closes the Dynamic Data Exchange (DDE) channel.

Example

```
DDETerminate $CHANNEL
```

Sample Script

```
* The following script starts Microsoft Excel with a test spreadsheet.
* "some data" is then put into row 1 column 1 of the spreadsheet.
* The spreadsheet is saved and Excel is closed.
* The DDE conversation is terminated.
*
Run "C:\Program Files\Microsoft Office\Office11\excel.exe
        ""D:\Test.xls"""
Wait 1000
DDEInitiate  $CHANNEL,"EXCEL","Sheet2"
DDEPoke      $CHANNEL,"R1C1","some data"
```

```
DDEExecute   $CHANNEL,"[SAVE()][QUIT()]"
DDETerminate $CHANNEL
exit
```

See Also:

DDEInitiate
DDEExecute
DDEPoke
DDERequest

6.3.5.14 EditSelect

```
EditSelect   top,left,bottom,right
```

Where:

top	is the first row of the selection rectangle.
left	is the left column of the selection rectangle.
bottom	is the last row of the selection rectangle.
right	is the right column of the selection rectangle.

top, left, bottom and/or right may be variables.

The **EditSelect** command selects data in the specified rectangle. The selected data may be moved to the clipboard using the following commands:

```
Key EditCopy
key EditCopyAppend
Key EditCut
Key EditCutAppend
```

Examples

```
EditSelect     1,1,10,80
EditSelect     24,1,24,80
EditSelect     $TOP,$LEFT,$BOTTOM,$RIGHT
```

The following script copies screen data to the clipboard and then saves the data to a file.

```
EditSelect     1,1,5,80
key            EditCopy
FileSpec       clipboard,c:\ClipboardData.txt",append
key            EditSaveClipboard
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.15 exit

```
exit
```

The **exit** command terminates a script. The exit command has no parameters.

Example

```
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.16 FileSpec

```
FileSpec    {log | printer | clipboard, } "filename"[,append])
```

Where:

log	indicates a log file specification for the HostSave or HostEnableLogging system commands.
printer	indicates a printer file specification for the HostPrintScreen system command.
clipboard	indicates a file specification for the EditSaveClipboard system command.
filename	is the full filename, including drive letter and path name for the output file.
append	append output to an existing file.

filename may be a variable.

The **FileSpec** command defines the disk output files for TN3270 Plus system commands.

The "FileSpec log" command directs the terminal session activity to the specified output file when logging is enabled (HostEnableLogging). The HostSave system command output is also directed to the specified file. The HostSave system command can be issued with the script command "key HostSave."

The "FileSpec printer" command directs PrintScreen system command output to a disk file. The HostPrintScreen system command can be issued with the script command "key HostPrintScreen." To reset output to the printer use the following:

FileSpec printer,""

The "FileSpec clipboard" command directs EditSaveClipboard system command output to the specified file The EditSaveClipboard command can be issued with the script command "key EditSaveClipboard."

Examples

```
FileSpec log,"c:\test.log",append
FileSpec log,$FILENAME
FileSpec printer,"c:\printer.txt"
FileSpec clipboard,"d:\Clipboard.txt",append
```

The following script defines c:\test.log" as the current log file and then saves the current screen to the log file.

```
FileSpec   log,"c:\test.log"
key        HostSave
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.17 FileTransfer

PC to Host:

```
FileTransfer      Operation={send | $variable},
      PcFile={PC filename | $variable},
```

```
      HostFile={Host filename | $variable},
      Opsys={vm/cms | cics | mvs/tso | $variable},
      [Blksize={blksize | $variable},]
      [Lrecl={lrecl | $variable},]
      [Recfm={default | fixed | variable | undefined | $variable},]
      [Mode={WSF | ScreenImage | $variable},]
      [BufferSize={ buffsize | $variable},]
      [Language={language | $variable},]
      [Options={options | $variable},]
      [Program={program | $variable}]
```

Host to PC:

```
FileTransfer      Operation={receive | $variable},
      PcFile={PC filename | $variable},
      HostFile={Host filename | $variable},
      Opsys={vm/cms | cics | mvs/tso | $variable},
      PCOptions={replace | append | prompt | $variable},
      [Mode={WSF | ScreenImage | $variable},]
      [BufferSize={ buffsize | $variable},]
      [Language={language | $variable},]
      [Options= {options | $variable},]
      [Program={program | $variable}]
```

Where:

$variable	is a variable. The text defined for the variable is substituted for the variable name. A variable may be used in place of any of the FileTransfer command keyword parameters.
send	send a file from the PC to the session host computer (upload). This parameter is not case sensitive.
receive	move a file from the session host computer to the PC (download). This parameter is not case sensitive.
PC filename	is the full filename, including drive letter and path, of a PC file. Enclose the PC filename in double quotation marks ("") if it contains embedded spaces or commas.
Host filename	is the name of a file on the host computer. Enclose the Host filename in double quotation marks ("") if it contains embedded spaces or commas.
vm/cms | cics | mvs/tso	is the operating system on the host computer. Specify "vm/cms", "cics" or "mvs/tso."
blksize	is the block size for the transferred file. A valid blksize is between 1 and 32760. (MVS/TSO send operations only)
lrecl	is the logical record length for the transferred file. A valid lrecl is between 1 and 32760. (Send operations only)
default | fixed | variable | undefined	is the record format for the host file. Specify one of the following: "default", "fixed", "variable" or "undefined". (Send operations only)

default	use the default record format for the host system.
fixed	indicates fixed length records.
variable	indicates variable length records.

undefined indicates the record format is not defined.

replace | append | prompt is one of the following options: "replace", "append" or "prompt". (Receive operations only)

- replace — if the PC file already exists, replace it with the download file.
- append — if the PC file already exists, add the download file to the end of the existing file.
- prompt — if the PC file already exists, prompt the user to replace the file or cancel the transfer operation.

WSF | ScreenImage is the type of file transfer. WSF indicates a Write Structured Field file transfer with the buffer size specified in the BufferSize parameter. Screen images indicates a screen image file transfer. The default is WSF.

buffsize is the buffer size for WSF file transfers. A valid buffsize is between 512 and 65535. The default is 2048.

language specify the name of the host language code page (*.cpg) to use for this file transfer operation. If a language file is not specified, the host language code page defined for the session is used by default.

options specify other IND$FILE options. The options in this parameter are passed directly to IND$FILE exactly as they appear. There is no editing of the options in this field. This allows you to pass options specific to your version of IND$FILE. Enclose the options string in double quotation marks ("") if it contains embedded spaces or commas. Some common options are:

- ascii — Converts ASCII to EBCDIC when the PC sends a file to the host, and converts EBCDIC to ASCII when the PC receives a file from the host.
- crlf — Replaces the carriage return/line feed characters with line breaks when the PC sends a file to the host, and replaces line breaks with carriage return/line feed characters when the PC receives a file from the host.
- append — Appends the transferred file to the host file. (Send operations only.)
- notrunc — (z/OS (MVS) and VM) Do not truncate trailing blanks.
- blank — (CICS) Do not truncate trailing blanks.

z/OS (MVS) file transfers

space(pri [,sec]) tracks | cylinders | avblock(size)
specify the space allocation for a new data set (Send only). Where:
pri is the primary allocation.
sec is the secondary allocation.
size is the number of bytes in an avblock.
tracks | cylinders | avblock(size) indicates the unit of allocation.

Examples:

```
space(15,1) tracks
space(2,1) cylinders
space(15,1) avblock(1024)
```

VSE file transfers

- file=rdr | lst | pun — specify the POWER queue location for the host file. (VSE file transfers only)
- file=ts — specify the host file location is in CICS/VSE temporary storage. (VSE file transfers only)
- file=lib — specify the host file is in a VSE library. (VSE file transfers only)
- l=libname — specify the name of the VSE library. (VSE file transfers only)
- s=sublib — specify the name of the VSE sublibrary. (VSE file transfers only)

program specify the file transfer program name. By default the program name is IND$FILE.

The **FileTransfer** command initiates a file transfer. File transfer operations are only valid for 3270 sessions.

Examples

```
FileTransfer   operation=send,
         pcfile=e:\download\testfile.txt,
         hostfile="testfile upload a",
         opsys=vm/cms,
         lrecl=80,
         recfm=fixed,
         options="ascii crlf"

FileTransfer   operation=receive,
         pcfile=e:\download\testfile.txt,
         hostfile="testfile upload a",
         opsys=vm/cms,
         pcoptions=prompt, o
         options="ascii crlf"

FileTransfer   operation=receive,
         pcfile=e:\test.txt,
         hostfile="test file a",
         opsys=vm/cms,
         pcoptions=prompt,
         language="C:\Program Files\SDI\TN3270 Plus\france.cpg",
         options="ascii crlf"

FileTransfer   operation=send,
         pcfile=e:\test.txt,
         hostfile='APPL.TEST.DSN(TEST)',
         opsys=mvs/tso,
         options="ascii crlf"

FileTransfer   operation=receive,
         pcfile=e:\test.txt,
         hostfile='APPL.TEST.DSN(TEST)',
         opsys=mvs/tso,
         mode=wsf, buffersize=65535,
         pcoptions=prompt,
         options="ascii crlf"
```

The following script transfers file "profile exec a" from the VM/CMS host to "C:\download\profile.txt" on the PC.

```
FileTransfer   operation=receive,
         pcfile=c:\download\profile.txt,
         hostfile="profile exec a",
         opsys=vm/cms, pcoptions=prompt,
         options="ascii crlf"
exit
```

The following script sends file "C:\download\profile.txt" on the PC to "test exec a" on the VM/CMS host.

```
FileTransfer   operation=send,
         pcfile=c:\download\profile.txt,
         hostfile="test exec a",
         opsys=vm/cms,
         lrecl=80,
         recfm=default, options="ascii crlf"
exit
```

See Also:

Script Command Summary

Script Variables
Using the Scripting Language

6.3.5.18 find

```
find  $variable,stringToSearch,stringToFind[,start]
```

Where:

$variable	receives the position, relative to 1, where stringToFind first appears in stringToSearch . If the string is not found, $variable contains 0 (zero).
stringToSearch	the string to be searched. Enclose stringToSearch in double quotation marks ("") if it contains embedded spaces or commas.
stringToFind	the string to find in stringToSearch. Enclose stringToFind in double quotation marks ("") if it contains embedded spaces or commas.
start	an optional parameter that specifies the position, relative to 1, within stringToSearch where the find operation should begin.

stringToSearch, stringToFind and start may be variables.

The **find** command finds a string within another string and returns the position of the found string in a variable.

Examples

```
find     $Position,"The quick brown fox",fox
find     $Position,$string1,$string2,10
find     $Position,$SCREEN,Login:
```

6.3.5.19 GetField

```
GetField $variable,string,field#[,separator]
```

Where:

$variable	receives the field read from the string.
string	is the string containing the field to be extracted.
field#	is the field number to extract.
separator	is the single character field separator. The default is a comma. If coding a comma it should be enclosed in double quotes.

string, field# and separator may be variables.

The **GetField** command extracts a field from a field-separated string and copies it into a script variable.

Examples

```
GetField       $FIELD,"The quick brown fox",2," "
GetField       $FIELD,"The,quick,brown,fox",2,","
```

Sample Script

```
* GetField.txt
* Sample script to extract fields
*
SetVar $String,"The quick brown fox"
SetVar $FieldNumber,1
SetVar $Fields,4

LOOP:
GetField $Field,$String,$FieldNumber," "
type $Field
CalcVar $FieldNumber,$FieldNumber,+,1
```

```
if $FieldNumber,LE,$Fields,LOOP:

Exit
```

6.3.5.20 GetString

```
GetString $variable,filename,line#[,label:]
```

Where:

$variable	receives the text read from the file.
filename	is the full filename, including drive letter and path name for the input file. This file must be a text file. Enclose the filename in double quotation marks ("") if it contains embedded spaces or commas.
line#	is the line number of the line to be read from the input file.
label	is the label of an error handler routine. The script branches to this routine if the I/O fails. The I/O error handler routine can use the $FILESTAT variable to get the cause of the error. If this parameter is omitted, the user will be prompted with the script error dialog box which allows the command to be retried or ignored.

filename and line# may be variables.

The **GetString** command reads a line of text from a file and copies it into a script variable.

Examples

```
GetString       $LINE,c:\test.txt,1
GetString       $LINE,c:\test.txt,1,IOError:
GetString       $Line,$FileName,$LineNumber
```

Sample Script

```
* GetString.txt
* Sample script to read a file
*
SetVar $FileName,f:\script\sample.txt
SetVar $LineNumber,1
SetVar $Lines,4

LOOP:
GetString $Line,$FileName,$LineNumber
type $Line
key "enter"
CalcVar $LineNumber,$LineNumber,+,1
if $LineNumber,LT,$Lines,LOOP:

Exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language
PutString Command

6.3.5.21 global

```
global  $var1[,$var2] ... [,$varn]
```

Where:

$varn	is a variable. The variable is available to scripts called by using the include script

command.

The **global** command defines global variables. The data in global variables is available to scripts called using the include script command. Local variables are not available to scripts called using the include command.

Examples

```
global   $NAME
global   $USERID,$PASSWORD
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.22 goto

```
goto  label:
```

Where:

label	is a label in the script. label names are case sensitive, contain no embedded blanks and end with a colon. This parameter may be a variable containing a label name.

The **goto** command branches to the specified label. Labels are defined by placing a label name on a script line starting in column one.

Examples

```
goto TEST_LABEL:
goto LABEL1:
goto $LABELNAME
```

The goto command in the following script branches to label "EXIT:", skipping the two commands between "LABEL2:" and "EXIT:".

```
WaitFor  RECON,LABEL1:,Ready,LABEL2:
LABEL1:
type     begin
key      enter
goto     EXIT:
LABEL2:
type     test
key      enter
EXIT:
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.23 if

```
if $var,{EQ | NE | LT | GT | GE | LE},{text | $var},label:[,NOCASE]
```

Where:

$var	is a variable. The text defined for the variable is substituted for the variable name and used in the comparison.

EQ | NE | LT | GT | GE | LE		is the comparison type.
	EQ	equal
	NE	not equal
	LT	less than
	GT	greater than
	GE	greater than or equal to
	LE	Less than or equal to
text		is any text string. Enclose the text in double quotation marks ("") if it contains embedded spaces, tabs or commas.
label:		is a label in the script to branch to if the comparison is true. label names are case sensitive, contain no embedded blanks and end with a colon.
NOCASE		use a comparison that is not case sensitive.

The **if** command branches to the specified label in the script if the comparison is true. The comparisons in the if statement are string comparisons.

Examples

```
if $name,EQ,"John",ProcessName:
if $SCREEN(1,1,1,4),EQ,"John",ProcessName:,NOCASE
if $CLIPBOARD,LT,"225",ProcessCount:
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.24 include

```
include  filename
```

Where:

filename	is the full filename, including drive letter and path name, of a script file. Enclose the script filename in double quotation marks ("") if it contains embedded spaces or commas. This parameter may be a variable containing a script filename.

The **include** command calls another script file. All the commands in the called script file are executed, and then processing continues in the calling script. An included script file may include additional script files. There is no limitation on how deeply script files can be nested.

If you have a group of commands that are used in many different scripts, put the group of commands into a separate script file and include that script file in place of the actual commands. Variables defined in the calling script are not available to the included script unless they are defined using the global script command.

Examples

```
include  c:script.txt
include  c:\scripts\script.txt
include  $FILENAME
```

The following script calls scripts to logon 2 users.

```
include  "c:\scripts\logon user 1.txt"
include  "c:\scripts\logon user 2.txt"
exit
```

See Also:

Script Command Summary

Script Variables
Using the Scripting Language

6.3.5.25 key

```
key  {keyname | command | $variable}
```

Where:

keyname	is the name of the key on the host keyboard.
command	is the name of a system command. This parameter allows TN3270 Plus menu commands to be issued from within a script.
$variable	is a variable. The text defined for the variable is substituted for the variable name and used as the command parameter.

The **key** command specifies the key to be simulated. The complete list of keynames can be found in the **Keyboard Map Setup** dialog box. To display the **Keyboard Map Setup** dialog box, open the **Setup** menu, select **Sessions...** from **Setup Items** choose **Keyboard** and then click the **Configure...** button. Select Terminal Keys in the **Function Group** drop-down list box.

You may also consult the default keyboard maps in this help file:

Default 3270 Keyboard Map
Default 5250 Keyboard Map
Default VT100/VT220 Keyboard Map

For VT100, VT220 and ANSI terminal emulation sessions, you can also send "Ctrl+A" through "Ctrl+Z" . For example, to send "Ctrl+A" you would use the following command. Note that the "Ctrl+A" is case sensitive.

```
key  Ctrl+A
```

The **key** command can also issue a system command from within a script. The complete list of system commands can be found in the **Keyboard Map Setup** dialog box. To display the **Keyboard Map Setup** dialog box, open the **Setup** menu, select **Sessions...** from **Setup Items** choose **Keyboard** and then click the **Configure...** button. Select System Commands in the **Function Group** drop-down list box.

Examples (3270)

```
key  enter
key  tab
key  PA2
key  PF1
key  $KEY
key  HostExit        *Exit TN3270 Plus
```

Examples (5250)

```
key  enter
key  tab
key  RollDown/PageUp
key  F1
```

Examples (VT100/VT220)

```
key  enter
key  tab
key  PageUp
key  F1
key  Ctrl+A
```

Examples (System Commands)

```
key      EditCopy              Copy selected data to the clipboard
key      EditPaste             Paste data into a session
key      EditSelectionLeft     Select the character left of the cursor
key      HostClose             Close the session
```

```
Key      HostEnableLogging       Enable terminal session activity
logging
key      HostExit                Exit TN3270 Plus
key      HostPrintScreen         Print the current screen
```

The following script types "this is a test" and then "presses" the enter key.

```
type     "this is a test"
key      enter
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.26 MsgBox

```
MsgBox   "message text"[[,icon][,"caption"]]
```

Where:

message text	is the text to display in the message dialog box. This parameter may be a variable containing the message text.
icon	is one of the following Windows msgbox icons: ICONSTOP, ICONEXCLAMATION, ICONQUESTION, ICONINFORMATION. If omitted ICONEXCLAMATION is used. This parameter may be a variable containing the name of the ICON to use.
caption	is the caption for the message box. If omitted the caption will be "TN3270 Plus". This parameter may be a variable containing the message box caption.

The **MsgBox** command pauses the script and displays a dialog box that contains a message for the user. The **MsgBox** dialog box displays the message text and **OK** and **Cancel** buttons. Click the OK button to continue script processing. Click the Cancel button to cancel the script.

Example

```
MsgBox   "The script completed successfully!",ICONINFORMATION
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.27 option

```
option   [option1][,option2]
```

Where:

option1	is one of the options listed below.
option2	is one of the options listed below.

Valid options are:

AskForPos(top,left)	specifies the position of the top left corner of the AskFor dialog box. top and left are the offset in pixels from the top left corner of the TN3270 Plus main window. The offset may be negative to move the dialog box outside the TN3270 Plus window.
AskForPos(center)	specifies that the AskFor dialog box be positioned in the center of the TN3270 Plus main window.
WaitForPos(top,left)	specifies the position of the top left corner of the WaitFor dialog box. top and left are

	the offset in pixels from the top left corner of the TN3270 Plus main window. The offset may be negative to move the dialog box outside the TN3270 Plus window.
WaitForPos(center)	specifies that the WaitFor dialog box be positioned in the center of the TN3270 Plus main window.

The **option** command specifies one or more options for a script. Options remain in effect until another option command is processed or to the end of the script.

Examples

```
option AskForPos(50,50)
option AskForPos(center),WaitForPos(center)
option AskForPos(0,-100)
option WaitForPos(50,50)
option WaitForPos(center)
```

6.3.5.28 PutString

```
PutString "text",filename,{create | append}[,label:]
```

Where:

text	is the text to write to the file.
filename	is the full filename, including drive letter and path name for the output file. This file must be a text file. Enclose the filename in double quotation marks ("") if it contains embedded spaces or commas.
create	Create a new file. If a file with the same name already exits, it is deleted.
append	Append text to an existing file. If the file does not exist, it is created.
label	is the label of an error handler routine. The script branches to this routine if the I/O fails. The I/O error handler routine can use the $FILESTAT variable to get the cause of the error. If this parameter is omitted, the user will be prompted with the script error dialog box which allows the command to be retried or ignored.

text and filename may be variables.

The **PutString** command writes a line of text to a file.

Examples

```
PutString      "This is a test",c:\test.txt,create
PutString      "This is a test",c:\test.txt,create,IOError:
PutString      $SCREEN(1,80),$FileName,append
```

Sample Script

```
 * PutString.txt
* Sample script to write to a file
*
SetVar $FileName,f:\script\PutStringFile.txt
SetVar $Position,1
SetVar $Columns,80
SetVar $MaxPosition,1920

* Write Screen contents line by line to the file
LOOP:
PutString $SCREEN($Position,$Columns),$FileName,append
CalcVar $Position,$Position,+,$Columns
if $Position,LT,$MaxPosition,LOOP:

Exit
```

See Also:

Script Command Summary

Script Variables
Using the Scripting Language
GetString Command

6.3.5.29 replace

```
replace  $variable,oldString,newString
```

Where:

$variable	variable containing text to be replaced..
oldString	the string to replaced. Enclose oldString in double quotation marks ("") if it contains embedded spaces or commas.
newString	the replacement string. Enclose newString in double quotation marks ("") if it contains embedded spaces or commas.

oldString and newString may be variables.

The **replace** command replaces all occurrences of oldString in $variable with newString.

Example

```
replace  $variable,dog,cat
```

Sample Script

```
* replace.txt
* replace fox with dog

SetVar $Result,"The quick brown fox"
SetVar $Old,"fox"
SetVar $New,"dog"
replace $Result,$Old,$New
type $Result
exit
```

6.3.5.30 run

```
run  filename{,HIDDEN | MINIMIZED | MAXIMIZED}
```

Where:

filename	is the full file name, including drive letter and path, of an application or batch file. Enclose the filename in double quotation marks ("") if it contains embedded spaces or commas. You can append command line parameters to the filename on the Run script command (within the double quotes, separated by a space). filename may be a variable containing the full filename of an application or batch file.
HIDDEN	run the application with its window hidden.
MINIMIZED	run the application with its window minimized.
MAXIMIZED	run the application with its window maximized.

filename may be a variable.

The **run** command runs the specified application or batch file. filename may specify a .com, .exe, .bat or .pif file.

Examples

```
run      "c:\batch\tasks job.bat"
run      c:\skey.exe,MINIMIZED
run      $PROGRAM,HIDDEN
run      "C:\Program Files\SDI\TN3270 Plus\tn3270.exe mainframe"
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.31 RunDirectory

```
RunDirectory   directory
```

Where:

directory	is a directory. Enclose the directory in double quotation marks ("") if it contains embedded spaces or commas. directory may be variable containing a directory name.

The **RunDirectory** command specifies the working directory for the application started by the **run** command. This is useful for applications that use a working directory for their files.

Examples

```
RunDirectory   "c:\program files\data"
RunDirectory   c:\test\data
RunDirectory   $DIRECTORY
```

The following script starts the program test.exe with a working directory of c:\program\data.

```
RunDirectory   c:\program\data
Run            c:\program\test.exe
exit
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.32 session

```
session  {session number | $variable}
```

Where:

session number	is the session number. Session numbers 1 to 99 are valid. session number may be a variable containing a valid session number.

The **session** command switches sessions while a script is processing. A script continues processing in the same session until a **session** command switches the script to another session.

Examples

```
session  1
session  2
session  $SESSION
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.33 SetPrt

```
SetPrt  {"printer" | "Application Default" | "RESTORE"}[,{P | L}]
```

Where:

printer	is the name of the printer.
Application Default	specifies application default printer. (Host, Print Setup...).
RESTORE	restore the original session printer setting.
P	sets the printer orientation to Portrait. Portrait is the default.
L	sets the printer orientation to Landscape.

The **SetPrt** command changes the session printer.

Examples

```
SetPrt          "HP Printer 1",L
SetPrt          "Application Default",P
SetPrt          "RESTORE"
```

Sample Script

The following script sets the printer to "HP Printer 1" with landscape orientation, prints the terminal screen and restores the session back to its default printer.

```
SetPrt          "HP Printer 1",L
key             HostPrintScreen
SetPrt          "RESTORE"
exit
```

6.3.5.34 SetVar

```
SetVar  $variable,{"string1" | $var1}[,{"string2" | $var2}...]
```

Where:

$variable	is the name of the variable to be set.
stringn	is a string of characters. stringn is case sensitive. Enclose stringn in double quotation marks ("") if it contains embedded spaces or commas.
$varn	is a variable. The text defined for the variable is substituted for the variable name and used as the command parameter.

The **SetVar** command initializes a variable to the specified text string or variable. If multiple stringn and/or $varn parameters are used they are concatenated together to form a new **$**variable.

Use the variable in place of the text in any script commands that accept a variable as a parameter.

Examples

```
SetVar   $NAME,"Sam Spade"
SetVar   $firstname,Sam
SetVar   $lastname,Spade
SetVar   $NAME,$firstname,"  ",$lastname
SetVar   $Name,$SCREEN(1,1,1,8)
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.35 SSLConnect

```
SSLConnect  {SSLv2 | SSLv3 | TLSv1 | $variable}
```

Where:

SSLv2	is SSL version 2
SSLv3	is SSL version 3
TLSv1	is TLS version 1
$variable	is a variable. The text defined for the variable is substituted for the variable name and used as the command parameter.

Use the SSLConnect command to begin a TLS or SSL connection when the host computer resides behind a proxy server or firewall.

If the host is behind a proxy server and an SSL connection is required then the connection must be made using a script instead of specifying SSL in the Host pane of the **Session Setup** dialog box. The reason for this is that if the SSL connection is made in the normal manner (using the radio buttons in the Host pane of the **Session Setup** dialog box) then the SSL handshake will take place with the proxy server instead of the host. Insert the **SSLConnect** command at the appropriate place in the script to cause the SSL connection to be made.

Examples

```
SSLConnect  SSLv3
SSLConnect  TLSv1
```

Sample Script

The following is a sample using the SSLConnect command.

```
*           Proxy Server login
WaitFor     username
type        john
key         enter
WaitFor     password
type        smith
key         enter
*           Login complete, connect to host
waitfor     "Login complete"
type        connect 127.0.0.1
key         enter
*           Initiate SSL handshaking
SSLConnect  TLSv1
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.36 trim

```
trim  $variable,[trimCharacter,[,LEFT | RIGHT]]
```

Where:

$variable	variable containing text to be trimmed.
trimCharacter	the leading and/or trailing character(s) to trim from $variable. If this parameter contains multiple characters all of the occurrences of these character will be trimmed. If trimCharacter is not specified, then it defaults to a space character. This parameter may be a variable containing the trim character or characters. Enclose trimCharacter in double quotation marks ("") if it contains spaces or commas.
LEFT	trim leading characters from $variable. If neither LEFT or RIGHT is specified, then both leading and trailing characters are trimmed.
RIGHT	trim trailing characters from $variable.

The **trim** command trims leading and/or trailing characters from a variable.

Example

```
trim     $Text
trim     $Text,0,left
trim     $Text," ", right
trim     $TEXT,"$ "
```

Sample Script

```
* trim.txt
*

SetVar $Text,"985.2000000"
trim $Text,"0",Right
type $Text
exit
```

6.3.5.37 type

```
type  "string1"["string2"["string3"...]]
```

Where:

stringn	is a string of characters to be typed into the session. This field is case sensitive. Enclose stringn in double quotation marks ("") if it contains embedded spaces or commas. If a double quote (") needs to be included in stringn enter two double quotes (""). The two double quotes will be replaced by a single double quote when the string is typed. Any of the stringn parameters may be variables. If stringn is a variable the text associated with the variable is substituted and typed into the session at the current cursor location.

The **type** command enters the string into a session at the current cursor location. If multiple stringn parameters are specified they are concatenated and typed into the session.

Examples

```
type     "query names"
type     $NAME
type     "This is how to include ""double quotes"" in the text"
type     $CLIPBOARD
type     $FIRSTNAME," ",$LASTNAME
```

Sample Script

The following script enters the "query names" command into a session.

```
type     "query names"
key      enter
exit
```

See Also:

Script Command Summary

Script Variables

Using the Scripting Language

6.3.5.38 wait

```
wait  milliseconds
```

Where:

milliseconds	is the number of milliseconds script processing should wait before continuing with the next command. milliseconds may be a variable containing the number of milliseconds.

The **wait** command pauses script processing for the specified number of milliseconds. If you wish to wait for a host response before continuing script processing, the **wait** command may not be the best choice. Since response times are difficult to predict and inconsistent, the wait must be long enough for the longest possible response. This is not very efficient. It may be better to use WaitFor or AskFor to pause the script.

Examples

```
wait  1000
wait  2000
wait  $TIME
```

See Also:

Script Command Summary
Script Variables
Using the Scripting Language

6.3.5.39 WaitFor

```
WaitFor  string1[,label1:] [,string2,label2:...,string10,label10:]
      [,minimized | hidden]
      [,WindowPos(top,left)]
```

Where:

stringn	is a string of characters. stringn is case sensitive. Enclose stringn in double quotation marks ("") if it contains embedded spaces or commas.
labeln	is the name of the label in the script. Labeln: is case sensitive.
minimized	minimize the WaitFor dialog.
hidden	hides the WaitFor dialog, so the user will not see it displayed.
WindowPos(top,left)	specifies the position of the top left corner of theWaitFor dialog box. top and left are the offset in pixels from the top left corner of the TN3270 Plus main window. The offset may be negative to move the dialog box outside the TN3270 Plus window. If this parameter is not specified, the dialog box is displayed in the center of the TN3270 Plus main window.

stringn, labeln, top and left may be variables

The **WaitFor** command accepts from 1 to 10 string,label: pairs.

WaitFor with a single parameter

Examples

```
WaitFor  USERID,minimized
WaitFor  ===>,hidden
WaitFor  $NAME
WaitFor  "Test text",WindowPos(10,10)
WaitFor  ""
```

When the **WaitFor** command is used with a single string parameter (no label), the **WaitFor** command repeatedly scans the session screen buffer for the specified string. When the string is found, script processing continues

with the next command. The **WaitFor** command displays a dialog box containing the scan string. If the scan cannot locate the string, the user can click the **Cancel Script** button in the dialog box to cancel the script or the **Cancel Scan** button to cancel the scan and continue with the next script command.

If the Waitfor command is specified with a null string parameter:

```
Waitfor  ""
```

the script waits until the next time the host updates the screen and then continues with the next command.

The purpose of this Waitfor command is to wait for a screen response from the host and then continue script processing. Use this command to prevent a script from issuing commands before the host is ready to receive them.

WaitFor with multiple parameters

Examples:

```
WaitFor  RECONNECTED,LABEL1:,Ready,LABEL2:
WaitFor  abcd,LA:,efgh,LE:,ijkl,LI:,mnop,LM:,qrst,LQ:,uvwx,LU:,yz,LY:
```

When the **WaitFor** command is used with multiple parameters, the **WaitFor** command repeatedly scans the session screen buffer looking for the string(s). When it finds one of the strings, it branches to the associated label. The purpose of this command is to wait for a screen response from the host and then take the appropriate action based upon what the host returns.

Warning:

The **WaitFor** command scans the entire screen buffer each time it is updated. If one of the strings in the **WaitFor** parameters is already on the screen when the **WaitFor** command is issued, the search is satisfied immediately. This defeats the purpose of waiting for a host response. Make sure any string you specify will not be on the screen when the **WaitFor** command is issued.

In the following script, the **WaitFor** command scans the session screen buffer for the strings "RECONNECTED" or "Ready". If the **WaitFor** command finds "RECONNECTED", it branches to "LABEL1:." If the **WaitFor** command finds "Ready", it branches to "LABEL2:."

```
WaitFor  RECONNECTED,LABEL1:,Ready,LABEL2:
LABEL1:
type     begin
key      enter
goto     EXIT:
LABEL2:
type     test
key      enter
EXIT:
exit
```

See Also:

- Script Command Summary
- Script Variables
- Using the Scripting Language

6.3.6 Script Variables (Release 3.5 and below)

The script language has four types of variables: built-in variables, Windows environment variables, local variables and global variables.

Built-in Variables

Built-in variables are predefined. All built-in variables start with a dollar sign ($) and are all upper case. The following built-in variables are available.

$CLIPBOARD	replaced by any text on the Windows clipboard.
$COLS	replaced by the number of screen columns.
$COMPUTERNAME	computer name.
$CONVERT(screen_position, {ROW | COL})	returns the row or column for the given screen position. SetVar $CursorRow,$CONVERT($CURSOR,ROW) SetVar $CursorColumn,$CONVERT($CURSOR,COL)
$CONVERT(row,col,POS)	returns the screen position relative to 1 for the given row and column. SetVar $ScreenPosition,$CONVERT(24,80,POS)
$CRLF	replaced by a carriage return and line feed.
$CURSOR	replaced by the cursor position on the screen relative to 1 (row 1 column 1).
$DATE	date in the local date format.
$DATEDMY	date in DD/MM/YYYY format.
$DATEMDY	date in MM/DD/YYYY format.
$DATEYMD	date in YYYY-MM-DD format.
$FILEERROR	the return code of the last script file I/O operation.

$FILESTAT(file_path, file_mode)	Returns an integer which is the status returned from an open request on the file. The file_mode should be one of the following: modeRead modeWrite modeReadWrite The returned integer will be one of the following standard Windows file exceptions: 0 = none. No error occurred. 1 = genericException. An unspecified error occurred. 2 = fileNotFound. The file could not be located. 3 = badPath. All or part of the path is invalid. 4 = tooManyOpenFiles. The permitted number of open files was exceeded. 5 = accessDenied. The file could not be accessed. 6 = invalidFile. There was an attempt to use an invalid file handle. 7 = removeCurrentDir. The current working directory cannot be removed. 8 = directoryFull. There are no more directory entries. 9 = badSeek. There was an error trying to set the file pointer. 10 = hardIO. There was a hardware error. 11 = sharingViolation. SHARE.EXE was not loaded, or a shared region was locked. 12 = lockViolation. There was an attempt to lock a region that was already locked. 13 = diskFull. The disk is full. 14 = endOfFile. The end of file was reached. $FILESTAT is normally used in an if statement. For example: If $FILESTAT("D:\test.txt",modeRead),EQ,2,FileNotFound: If $FILESTAT("D:\test.txt",modeWrite),EQ,11,FileAlreadyOpen:
$LEFT(string,length)	Returns the specified number of characters from the left side of the string. Use this variable to extract characters from a string or another variable. string and/or length may be a variable.
$LEN(string)	Returns the length of the string. string may be a variable.
$LONGDATE	date in local long date format.
$MID(string,start[,length])	Returns the specified number of characters from the string starting with the character number specified in the start parameter. string, start and/or length may be a variable.
$OIA	replaced by the text in the Operator Information Area line on the terminal screen
$RIGHT(string,length)	Returns the specified number of characters from the right side of the string. Use this variable to extract characters from a string or another variable. string and/or length may be a variable.
$ROWS	replaced by the number of screen rows.

$SCREEN[(start[,length])]	replaced by the text at the specified location on the terminal screen. where: start — is the starting position on the screen. Specify 1 to indicate row 1 column 1. length — is the number of characters to include. start and/or length may be a variable. The ((start[,length])) parameter is optional. If you specify "$SCREEN" with no parameter, it is replaced by the text contents of the entire screen. If the (start[,length]) parameter is invalid, no substitution will take place and the variable will be treated as a literal.
$SCREEN[(top,left,bottom,right[,RECT])]	replaced by the text at the specified location on the terminal screen. where: top — is the first row of the selection rectangle. left — is the left column of the selection rectangle. bottom — is the last row of the selection rectangle. right — is the right column of the selection rectangle. RECT — is optional and specifies that the selection area is a rectangle, not a string. top, left, bottom, right and/or RECT may be a variable. The (top,left,bottom,right) parameter is optional. If you specify "$SCREEN" with no parameter, the variable is replaced by the text contents of the entire screen. If the (top,left,bottom,right) parameter is invalid, no substitution will take place and the variable will be treated as a literal.
$SCREEN[(top,left,length)]	replaced by the text at the specified location on the terminal screen. where: top — is the first row of the selection rectangle. left — is the left column of the selection rectangle. length — is the number of characters to include. top, left and/or length may be a variable. The (top,left,length) parameter is optional. If you specify "$SCREEN" with no parameter, the variable is replaced by the text contents of the entire screen. If the (top,left,length) parameter is invalid, no substitution will take place and the variable will be treated as a literal.
$TIME	time in local time format.
$USERNAME	current logged on user name.

Windows Environment Variables

Windows environment variables can be used in script strings. Windows environment variables are enclosed in % signs and may be in upper or lower case. For example:

```
type "%COMPUTERNAME%"
```

Local Variables

Local variable names start with a dollar sign ($) and may be mixed case. Local variables are created dynamically by the AskFor and SetVar script commands which read text into a variable. Here are a couple of examples:

```
AskFor "Enter your userid and click OK",$NAME
SetVar  $NAME,"Sam Spade"
```

Most script commands accept a variable in place of literal data. The text contained in the variable is substituted for the variable when the script command is processed.

Global Variables

Global variable use the same syntax as a local variable. The variable name must start with a dollar sign ($) and may be mixed case. Global variables must be declared using the Global script command. The data in global variables is available within scripts called using the include script command. Local variables are not available to scripts called using the include command.

Variable Substring

You can extract a substring from a local variable, global variable or the built-in variable using the following format:

```
$variable[(start[,length])]
```

where:

start is the starting position within the variable.

length is the number of characters to include. length defaults to 1 if it is not specified.

start and/or length may be a variable.
For example, the following script types **Joh** on the screen.

```
SetVar $TEST,"John Doe"
type $TEST(1,3)
exit
```

Concatenated Variable Names

You can concatenate variable names. For example, $$SCREEN(1,4)$SCREEN(10,3) would result in a variable name $ followed by the contents of screen location 1-4 followed by the contents of screen location 10-12. Thus, if screen location 1-4 contained "WHIZ" and screen location 10-12 contained "KID" the result would be a variable name of "$WHIZKID".

See Also:

Script Command Summary
Using the Scripting Language

6.4 Tool and Status Bars

6.4.1 Standard Toolbar

The standard toolbar is displayed across the top of the application window, below the menu bar. The toolbar provides quick mouse access to many tools used in TN3270 Plus.

To hide or display the standard toolbar: Choose **Standard Toolbar** from the **View** menu (ALT, V, T).

Customize:

You can customize the standard toolbar by opening the **Setup** menu and selecting **Standard Toolbar...** or right

clicking on the standard toolbar.


Connect to a host computer. Select the host computer in the Connect to Host dialog box.


Disconnect the current session from the host computer.


Save the screen image to the active log file.


Transfer a file to or from the host computer.


Run a script.


Turn terminal activity logging on or off. This button is optional and my be added using the **Customize Toolbar** dialog box (Setup, Toolbars…, Customize… button).


Delete selected data from the screen and move it to the clipboard.


Copy selected data from the screen to the clipboard.


Insert the contents of the clipboard at the current cursor location.


Switch into or out of Full Screen mode.


Print the terminal screen.


Select a new font.


Make the current font larger.


Make the current font smaller.


Switch to the session number on the button.

6.4.2 Keypad Toolbar

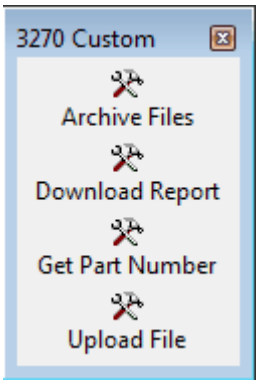
The keypad toolbar is a floating (or dockable) toolbar that contains the function keys for the current terminal emulation session. This toolbar provides quick mouse access to the terminal function keys.

To hide or display the keypad toolbar, choose **Keypad Toolbar** from the **View** menu (ALT, V, K).

- To change the keypad toolbar options (dockable, number of columns) choose **Toolbars...** from the **Setup** menu.
- You can customize the keypad toolbar by opening the **Setup** menu, selecting **Toolbar...** and clicking the keypad toolbar **Customize...** button, or by right clicking on the keypad toolbar.

6.4.3 Custom Toolbar

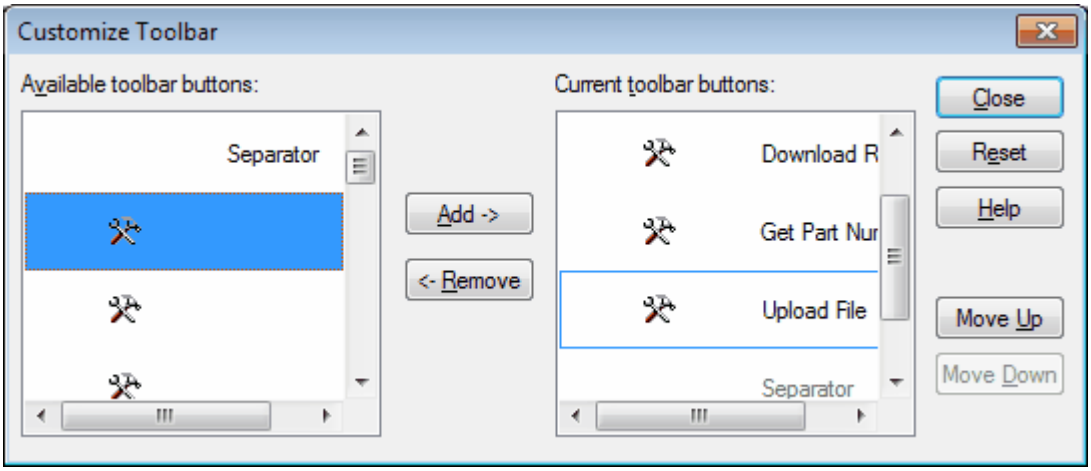
The custom toolbar is a floating (or dockable) toolbar that contains up to 20 buttons. These buttons can be assigned to commonly used scripts, macros and/or Windows commands. This gives you one-click access to commonly issued commands.



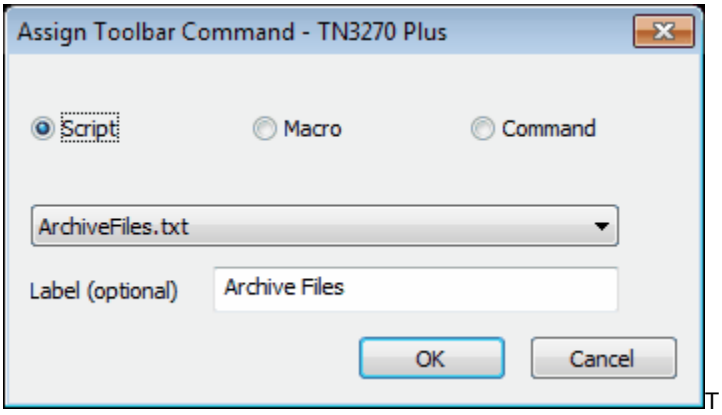
To hide or display the custom toolbar, open the **View** menu (ALT, V, K) and select **Custom Toolbar.**

To change the custom toolbar options (dockable, number of columns) open the **Setup** menu and select **Toolbars...**

To add buttons to, or remove buttons from, the custom toolbar, open the **Setup** menu, select **Toolbars...** and click the custom toolbar **Customize...** button.

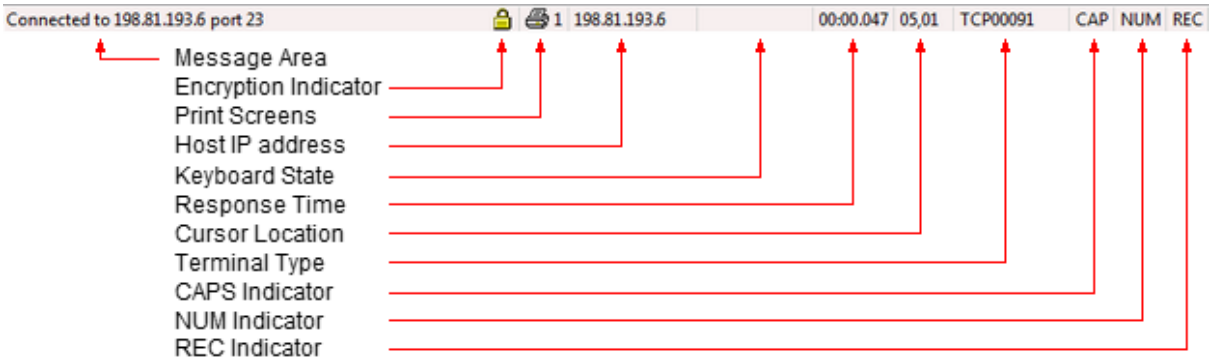


To assign a command to a toolbar button, right click on the toolbar button.



T

6.4.4 Status Bar



The status bar is displayed at the bottom of the TN3270 Plus window. To display or hide the status bar, use the **Status Bar** command in the **View** menu.

The following table describes each of the fields on the status bar.

Field	Description
Message Area	This area contains messages to help you use TN3270 Plus. A few of the uses follow: 1. Displays the URL address for the current connection. 2. Displays the line number of each line in a script as the script executes. 3. Describes actions of menu items as you use the arrow keys or the mouse to navigate through the menus.
Encryption Indicator	The padlock icon appears in this field for TLS or SSH connections. Double-click the padlock icon to display the server certificate detailsdialog box. Hover your mouse over the padlock to display the encryption method in use for the connection.
Print Screens	Indicates how many print screen are held in the print spool file when the "Print Screens to spool before printing" option is set.
Host IP address	The IP address of the host computer if it is available.
Keyboard State	Displays keyboard status. The following summarizes the keyboard status symbols: X *aid_key* — An AID generating key was pressed. X PROG*nnn* — Program check - a programming error was detected in the data from the host. X SYSTEM — The host system locked the keyboard. X <0> — Go Elsewhere - the cursor must be moved. It is in a protected field and an attempt has been made to enter, insert, erase or delete a character. X ?+ — Not accepted - the last input was not accepted X -f — Not available - the requested function is not available. X NUM — Numeric - an attempt was made to enter non-numeric data into a numeric-only field. MSG — Message waiting (5250 only)
Response Time	The elapsed time of the last command or function in **mm:ss.sss** format.
Cursor Location	Displays the cursor location in **row,column** format.
Terminal Type	Displays the terminal type. This is the terminal type used internally by telnet, this type is not always the same as the terminal name used by terminal manufacturers, so the terminal type may not match the name you selected in the **Connect to Host** dialog box.

CAPS Indicator	Displays "CAPS" when the Caps Lock key is latched down.
NUM Indicator	Displays "NUM" when the Num Lock key is latched down.
REC Indicator	Displays "REC" and blinks when a macro is being recorded.

6.4.5 Operator Information Area

The Operator Information Area (OIA) is displayed at the bottom of the TN3270 Plus terminal emulation area. To display or hide the OIA, use the **Operator Information Area** command in the **View** menu. The following table describes each of the fields in the OIA.

Field	Description
Connection status	Displays the system connection status. In the form: *tyz* Where: *t* is either "T" or "S" T indicates a TCP/IP insecure connection. S indicates a TCP/IP secure (TLS) connection. *y* is either "A" or "B". A Connected in non-TN3270E mode. B Connected in TN3270E mode. *z* is either "!" or "@" ! Working with SSCP (SSCP-LU mode). @ Working with host application (LU-LU mode).
Message Indicator	MSG - message waiting (5250 only)
Keyboard State	Displays keyboard status. The following summarizes the keyboard status symbols: X *aid_key* An AID generating key was pressed. X PROG*nnn* Program check - a programming error was detected in the data from the host. X SYSTEM The host system locked the keyboard. X <0> Go Elsewhere - the cursor must be moved. It is in a protected field and an attempt has been made to enter, insert, erase or delete a character. X ?+ Not accepted - the last input was not accepted X -f Not available - the requested function is not available. X NUM Numeric - an attempt was made to enter non-numeric data into a numeric-only field.
Session number	The session number of the currently displayed session.
Session name	The session name of the currently displayed session.
Insert mode indicator	Displays a caret (^) when in insert mode.
Cursor Location	Displays the cursor location in **row,column** format.
Response Time	The elapsed time of the last command or function in **mm:ss.sss** format.
Time	Displays the time in **hh:mm** format.
Terminal name | Date	If the session has an assigned terminal name, the terminal name is displayed. If the session does not have an assigned terminal name, the date is displayed in the system short date format.

7 Security Standards

7.1 Overview

There are numerous Security Standards that apply to today's Information Systems environment. We have added this Chapter as a reference to some of those standards.

7.2 Payment Card Industry Standard (PCI 3.0)

The PCI Security Standard Council offers security standards and supporting materials to enhance payment card data security. Compliance with the PCI Data Security Standard (PCI DSS) is considered vital for merchants that accept credit cards in an online or offline environment. TN3270 Plus helps organizations comply with these standards by offering the Secure Shell (SSH) and TLS (Transport Layer Security) encryption protocols.

For more information about the PCI Security Standards Council and their standards, please see the following links:

https://www.pcisecuritystandards.org/ - Home page

https://www.pcisecuritystandards.org/documents/PCI_DSS_v3.pdf - Payment Card Industry Data Security Standard Version 3.0 (PCI DSS 3.0)

8 FAQ (Frequently Asked Questions)

8.1 FAQ (Frequently Asked Questions)

Most Common Questions

Installation Questions

License Code Questions

Connections Questions

File Transfer Questions

General Questions

Most Common Questions and Answers

My disk has crashed and I have no backup

You need to purchase a new license for TN3270 Plus. Go to www.sdisw.com and click on the "Buy Now" link.

(Return to top)

Installation Questions and Answers

What causes "An error occurred during the move data process -113" during installation?

This error comes from InstallShield which we use to package our product. This message may appear when other applications are running while you are installing TN3270 Plus. You must close all other applications prior to installing TN3270 Plus. Close all other applications and try the install again.

If closing all other applications fails, the circumvention is to unzip the TN3270 Plus self-extracting file using a zip utility. Unzip the TN3270 Plus self-extracting file to a temporary folder and then execute the setup.exe program to start the installation wizard.

If this does not work for you, please send an e-mail to support@sdisw.com or fill out the Quick Contact Form. Please include your telephone number.

(Return to top)

What causes "An error occurred during the move data process -623" during installation?

This error comes from InstallShield which we use to package our product. This message appears when other applications are running while you are installing TN3270 Plus. You must close all other applications prior to installing TN3270 Plus. Close all other applications and try the install again.

If this does not work for you, please send an e-mail to support@sdisw.com or fill out the Quick Contact Form. Please include your telephone number.

(Return to top)

Error 1722.There is a problem with this Windows Installer package.

The full text of the message is: "Error 1722.There is a problem with this Windows Installer package. A program run as part of the setup did not finish as expected. Contact your support personnel or package vendor."

This message appears when one of the TN3270 Plus install routines is blocked by an antivirus program.

This message is normally caused by:

One of the Norton Antivirus applications (e.g. Norton Security Suite or Norton 360). The issue is caused by the Norton "SONAR Advanced Protection" routine. The problem can be resolved by turning off Norton SONAR Advanced Protection while TN3270 Plus is being installed.

(Return to top)

Can TN3270 Plus be installed on a network server?

Yes, TN3270 Plus can be installed on a network server and used by multiple users. See the License Agreement for licensing requirements and review "Using TN3270 Plus on Windows Terminal Server" in the "Administrators Guide..." section of the TN3270 Plus help file for instructions on configuration.

(Return to top)

Can TN3270 Plus be installed on Windows Terminal Server/Citrix MetaFrame?

Yes, TN3270 Plus runs on Windows Terminal Server and on Citrix MetaFame. TN3270 Plus configuration is handled in the same manner as when installing TN3270 Plus on a network server used by multiple users. See the License Agreement for licensing requirements and review "Using TN3270 Plus on an Intranet (Network)" in the "How to..." section of the TN3270 Plus help file for information on configuration.

(Return to top)

License Code Questions and Answers

The license code is not valid for this release.

The license code you entered is for an older release of TN3270 Plus. The license code is generated for a specific release and will not function with newer releases of TN3270 Plus.

The product release level is three digits separated by periods, for example, 3.4.0. The license code only checks the first two digits, so a license code created for 3.4.0 will work with releases 3.4.0 through 3.4.9. In addition, license codes for release 3.3 and above will work with the next higher release. For example, a release 3.4 license code will also work with release 3.5 (3.5.0 - 3.5.9). However, if you try to use a release 3.4 license code with release 3.6.0 or higher you will get the above error message.

If you have purchased a maintenance and support subscription, updates are free. Send an e-mail to sales@sdisw.com with your current "licensed to" name and the new release level and you will be e-mailed a license code for the new release.

If you do not have a maintenance and support subscription, you will need to subscribe to our maintenance and support plan in order to receive the new product release.

(Return to top)

The 'License code' field is incorrect.

The "License code" field contains a typographical error. Make sure that all the characters are correct.

(Return to top)

The 'Licensed to' field is incorrect.

The name in the "Licensed to" name does not match the name encrypted into the license code.

1. The "Licensed to" name is incorrect or contains a typographical error.
2. Make sure the "Licensed to" name has the correct number of spaces.
3. The "Licensed to" name and license code are issued in pairs. The "Licensed to" name must correspond to the license code you entered.

(Return to top)

Connection Questions and Answers

Can I connect to another PC running Windows?

Yes, it is possible use telnet to connect a Windows system to another Windows system and gain access to the command line. It is not possible to gain access to the Windows GUI. The requirements are as follows.

1. Both computers must be connected to the Internet or on the same network.
2. The computer you wish to connect to must be running a telnet server. Windows XP Professional and Windows 2000 Professional come with telnet servers. Use the following instruction to start the telnet server:

Windows XP Professional	Start, Control Panel, Administrative Services, Services, Double click on the Telnet service, on the General Tab specify the Startup type and then start the service.
Windows 2000 Professional	Start, Settings, Control Panel, Administrative Tools, Services, Double click on the Telnet service, on the General Tab specify the Startup type and then start the service.

   If the PC is running another version of Windows, you will have to purchase a telnet server application to run on the PC.

3. You must turn off NTLM authentication on the Windows system running the telnet server because TN3270 Plus does not support NTLM authentication.

Windows XP Professional	(Open a command prompt and enter `tlntadmn config sec=-ntlm`)

Windows 2000 Professional	(Open a command prompt and enter
	`tlntadmn`
	`3 *Display/change registry settings`
	`7 *NTLM`
	`Y *Do you want to change this value?`
	`0 *Turns NTLM off`
	`Y`
	`0`
	`0`

4. Connect TN3270 Plus. You need the IP address of the Windows PC running the telnet server. Use port 23 and terminal type VT100. You will be prompted for a username and password. The username and password must be defined on the Windows PC you are connecting to.

(Return to top)

How do I enter a telephone number into TN3270 Plus?

You cannot enter a telephone number into TN3270 Plus. TN3270 Plus does not support a direct dial connection to a host computer. TN3270 Plus connects to a host computer via TCP/IP using the Internet or an intranet as the communication link. As an example, SDI employees use TN3270 Plus to connect via the Internet to our IBM zSeries (mainframe). To connect:

1. The user establishes an Internet connection through their ISP.
2. Starts TN3270 Plus and enters the IP address of the mainframe in the Connect to Host dialog box and clicks the Connect button.
3. The connection is made across the Internet to TCP/IP running on the mainframe.

(Return to top)

Host www.xxx.yyy.zzz Is Unreachable

This message is normally caused by one of the following:

1. You are not connected to the Internet. You must make your connection to the internet before using TN3270 Plus to connect to a host computer. If you are an America Online (AOL) user, you must connect to AOL and login before trying to connect using TN3270 Plus.
2. There is a problem on the Internet and currently there is no route to the host computer. Try again later.
3. If the IP address in the message is incorrect, then you have the wrong "Host Name:" in the TN3270 Plus "Connect to Host" dialog box.
4. You have the incorrect port number specified in the "Connect to Host" dialog box.

(Return to top)

Socket error 0 (WSAEUNDEFINED)

Host www.xxx.yyy.zzz port nnn Is Unreachable: Socket error 0 (WSAEUNDEFINED)

You are attempting to connect to an Internet host address and your computer is not connected to the Internet.

(Return to top)

Socket error 10022 (WSAEINVAL)

Host *hostname* port *nnn* Is Unreachable: Socket error 10022: WSAEINVAL

This error is normally caused by one of the following:

1. The hostname specified in the Host Name edit box of the Connect to Host dialog box (Host, Connect...) is not valid.

   The Host Name should be the dotted IP address or the DNS name of the host computer. Every computer on the Internet has an IP address that looks something like this "128.228.1.20". To make this address easier to remember, the IP address may be registered and assigned a DNS name. For example, acmecomputer.com. When someone tries to connect to acmecomputer.com a DNS server on the Internet translates that to 128.228.1.20. In this example, either acmecomputer.com or 128.228.1.20 could be used as the host name.

   Think of the host name as the address of the host computer. It tells TN3270 Plus where the host computer is. You need to find out what the host name or IP address is for the computer you want to connect to. Try talking to the network administrator at the host computer site if you are unsure.

2. On an intranet (internal network), you may need to make an entry in the Windows "hosts" file to equate the hostname to the IP address.

   You will find the Windows hosts file here:

Windows 11/10/8/7/XP Window Server 2022/2019/2016/ 2012/ 2003,	c:\windows\system32\drivers\etc\hosts
Windows 2000	c:\winnt\system32\drivers\etc\hosts

   This file should contain an entry that looks something like this.

   192.168.1.5 *hostname*

   Where:

   hostname is the hostname from the error message.
   192.168.1.5 is the IP address of the host on your network.

(Return to top)

Socket Error 10051 (WSAENETUNREACH)

Session x (hostname:port) Socket Error 10051 (WSAENETUNREACH). A socket operation was attempted to an unreachable socket.

This usually means TN3270 Plus cannot find a route to reach the remote host.

This error is normally caused by an incorrect IP address in the "Host Name" edit box in the "Connect to Host" dialog box.

Try pinging the host computer to see if it is responding. You can this from the Windows command prompt (Start, All Programs, Accessories, Command Prompt) by entering "ping hostname" or "ping IP address" on the command line. If Ping fails, talk your network administrator and get the correct IP address.

(Return to top)

Socket Error 10060 (WSAETIMEDOUT)

Session x (*hostname:port*) Socket Error 10060 (WSAETIMEDOUT). A connection attempt failed because the connected party did not respond properly after a period of time, or established connection failed because connected host has failed to respond.

This message means that TN3270 Plus has sent communication to the host computer and the host computer did not respond before the end of the timeout period. The timeout period is monitored by the Windows socket, when the end of the timeout period is reached the Windows socket indicates the error to TN3270 Plus and TN3270 Plus reports the error.

This message is normally caused by one of the following:

1. An incorrect IP address or Host Name in the "Connect to Host" dialog box.

2. An incorrect port number specified in the "Connect to Host" dialog box.

3. A proxy server or firewall is blocking communication between TN3270 Plus and the host computer.

Try pinging the host computer to see if it is responding. You can this from the Windows command prompt (Start, All Programs, Accessories, Command Prompt) by entering "ping hostname" or "ping IP address" on the command line. If Ping times out, there is a problem at the host. Talk to a network administrator at the host site and see if they can help you.

(Return to top)

Socket Error 10061 (WSAECONNREFUSED)

Session x *(hostname:port*) Socket Error 10061 (WSAECONNREFUSED). No connection could be made because the target machine actively refused it.

This message is normally caused by one of the following:

1. The remote system rejected your attempt to connect with it, either because no server is listening on the specified port or it's unable to accept any additional connections. Verify you are using the correct port number and then contact the Network Administrator at the host computer site.

2. You have specified "localhost" as the Host Name in the Connect to Host dialog box. "localhost" is the host name of your PC. Replace localhost with the Host Name of the computer you wish to connect to.

   The Host Name should be the dotted IP address or the DNS name of the host computer. Every computer on the Internet has an IP address that looks something like this "128.228.1.20". To make this address easier to remember, the IP address may be registered and assigned a DNS name. For example, acmecomputer.com. When someone tries to connect to acmecomputer.com a DNS server on the Internet translates that to 128.228.1.20. In this example, either acmecomputer.com or 128.228.1.20 could be used as the host name. Think of the host name as the address of the host computer. It tells TN3270 Plus where the host computer is. You need to find out what the host name or IP address is for the computer you want to connect to. Try talking to the network administrator at the host computer site if you are unsure.

(Return to top)

Socket error 11001 (WSAHOST_NOT_FOUND)

Failed to create new socket: Socket error 11001 (WSAHOST_NOT_FOUND). No such host is known.

The name is not an official host name or alias, or it cannot be found in the database(s) being queried. This error may also be returned for protocol and service queries, and means that the specified name could not be found in the relevant database.

This message is normally caused by one of the following:

1. Specifying an incorrect host name in the "Host Name" edit box in the Connect to Host dialog box.
2. Specifying the incorrect IP version level in the Connect to Host dialog box. For example, specifying IPv6 for an IPv4 connection.

(Return to top)

Socket error 11004 (WSANO_DATA)

Failed to create new socket: Socket error 11004 (WSANO_DATA). No such host is known.

The requested host name is valid and was found in the database, but the associated data is not correct.

This message is normally caused by one of the following:

1. Specifying an incorrect host name in the "Host Name" edit box in the Connect to Host dialog box.
2. Specifying the incorrect IP version level in the Connect to Host dialog box. For example, specifying IPv6 for an IPv4 connection.

(Return to top)

Session x disconnected by host www.xxx.yyy.zzz.

This message is the result of a disconnect command received by TN3270 Plus.

This message is normally caused by one of the following:

1. The host computer disconnected you due to inactivity or technical problems.
2. If you are using an Internet connection

   a.) Your ISP disconnected you due to inactivity or technical problems.

   b.) You were disconnected from your ISP due to technical problems. For example, if you manually disconnect your ISP while TN3270 Plus is connected a host computer you will get the "Session x disconnected by host" message.

3. There is problem on a firewall, proxy server or router along the connection route.

(Return to top)

File Transfer Questions and Answers

INDFT018 Incorrect option specified: file transfer canceled

If you get the INDFT018 message when you are attempting to upload a new or replacement member to an existing PDS, then you must specify the following "Host File Options (PC to Host)" in the File Transfer dialog box:

- Default
- LRECL 0
- BLKSIZE 0

(Return to top)

Does TN3270 Plus have FTP support?

Yes. TN3270 Plus FTP is an optional, additional cost feature.

(Return to top)

General Questions and Answers

Can I display multiple sessions in multiple windows so that I can see more than one session at a time?

Yes, most users start multiple sessions in one TN3270 Plus window and use the session tabs to switch from one session to another. This does not allow you to view more that one session at a time. If you want to be able to see more than one session at a time, start TN3270 Plus multiple times. Each time you start TN3270 Plus you create a new window. You may then connect one or more a sessions in each window.

You can set the "Start each session in a new window" option (Setup, Preferences..., Start each session in a new window) to indicate you want each new session in a new window. TN3270 Plus will open a new Window for each new session you connect. You may also specify the "Start in a new window" option for specific sessions. (Setup, Sessions..., Setup Items = Host, Start in a new window").

(Return to top)

IKT00405I SCREEN ERASURE CAUSED BY ERROR RECOVERY PROCEDURE ***

The host sent a data stream that is not supported by 3278 terminal emulation. Change your terminal type from 3278 to 3279 (Host, Connect..., Advanced button, Setup Items = Terminal, Terminal Type = 3279).

(Return to top)

How do I setup TN3270 Plus so it connects to the same host every time it is started?

There are two ways to automatically start a session when TN3270 Plus is started:

1. Set the "Connect at Startup" option on for the session.
2. Include the session name on the TN3270 Plus command line.

"Connect at Startup" option:

1. Open the Setup menu and select Sessions.
2. In the Session Setup dialog box, click on the Connect at startup check box.
3. You may set this option on for multiple sessions and each session will connect when TN3270 Plus is started.

Include the Session Name on the Command Line:

You can include a session name on the command line when TN3270 Plus is started.

C:\Program Files\SDI\TN3270 Plus\TN3270.exe “*session_name*”

When the session name is included, TN3270 Plus connects to that session when it starts. For example:

C:\Program Files\SDI\TN3270 Plus\TN3270.exe “P/390 27x132”

Where:

“P/390 27x132” is the session name.

Tips:

- The session name is case sensitive.
- If the session name contains any spaces, it must be enclosed in double quotes as in the example above.

(Return to top)

Is there a Macintosh version of TN3270 Plus?

TN3270 Plus only supports Windows operating systems.

(Return to top)

How do I type the EBCDIC logical not sign (¬) into a terminal session?

You can assign any character to any keyboard and/or mouse button combination you wish. The following example assigns the logical not sign (¬) to the shift key plus the number 6 (shift+6).

1. Open the Setup menu and select Keyboard.

2. In the Session Setup dialog box, click the Configure... button.
3. In the Keyboard Setup dialog box, select "Characters" in the "Function Group" drop-down list box.
4. In the Function list box, scroll down to the logical not sign, "¬ (172, 0xAC)" and select it.
5. Click the Add Key... button.
6. In the Type Key dialog box, hold down the shift key and press the number 6. The edit box should display "shift+6."
7. Click the Retry button to correct an error and click the OK button to make the assignment.
8. In the Keyboard Setup dialog box, click the OK button.
9. In the Session Setup, dialog box, click the OK button.

(Return to top)

Is it possible to change the character assignment for a keyboard key? I want to make the decimal point key on the numeric keypad enter a comma instead of a decimal point.

You can assign any character to any keyboard and/or mouse button combination you wish. The following example assigns a comma to the decimal point key on the numeric keypad (Num.).

1. Open the Setup menu and select Keyboard.
2. In the Session Setup dialog box, click the Configure... button.
3. In the Keyboard Setup dialog box, select "Characters" in the "Function Group" drop-down list box.
4. In the Function list box, scroll down to the comma, ", (44, 0x2C)" and select it.
5. Click the Add Key... button.
6. In the Type Key dialog box, insure num lock is on and press the decimal point on the numeric keypad. The edit box should display "Num." (That is "Num" followed by a decimal point.)
7. Click the Retry button to correct an error and click the OK button to make the assignment.
8. In the Keyboard Setup dialog box, click the OK button.
9. In the Session Setup, dialog box, click the OK button.

(Return to top)

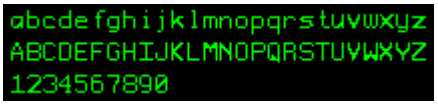
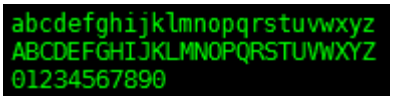
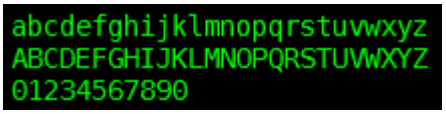
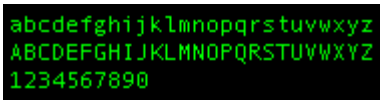
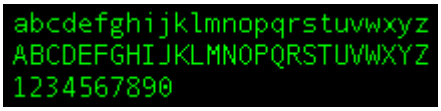
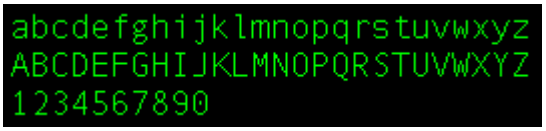
What screen fonts are available in TN3270 Plus?

TN3270 Plus can use any of the fixed-pitch fonts installed on your Windows system. In a fixed-pitch font, all the characters in the font are the same width. In a variable-pitch font, different characters have different widths. For example, a "W" is wider than an "i". A fixed-pitch font is required to align the characters in the columns on the emulated screen.

Try using Courier New with the style set to bold. This creates a clear readable screen on most monitors at most sizes. Select the font size in the font selection dialog (Setup, Display, Change... button) rather than resizing the terminal Window. TN3270 Plus may "stretch" or "compress" the font as you resize the Window. If the aspect ratio of the window is different than that of the font the font may look distorted. Choosing a specific font size avoids this problem.

Many users like the following fonts. They are free and work well with TN3270 Plus.

Font	Size	Sample TN3270 Plus display
ProFont bitmap	9 pt	
	11 pt	

	13 pt	
ProFont True Type	9 pt	
	11 pt	
	13 pt	
DejaVu Sans Mono True Type	9pt	
	11pt	
	13pt	
Andale Mono True Type	9 pt	
	11 pt	
	13 pt	

Let us know if you find other fonts you think work well with TN3270 Plus.

(Return to top)

Why doesn't my 3270 session display all the correct colors?

3278 terminal sessions support a limited selection of colors, just like real 3278 terminals. To get additional colors requires a 3279 terminal session. If you specified 3278 as your terminal type, change it to 3279 and you will get the additional colors that a 3279 terminal supports. (Host, Connect..., Advanced button, Setup Items = Terminal, Terminal Type = 3270).

(Return to top)

How do I make print screens print in "black and white" rather than in color or grayscale?

Setup, Printer..., Black and White Print Screen check box

(Return to top)

How do I start a printer session?

3287 printer support handles zSeries (mainframe) printer sessions. 5250 printer support handles iSeries (AS/400) printer sessions. Printer support is an optional, additional cost feature of TN3270 Plus and includes 3287 printer support, 5250 printer support and LPD.

To start a printer session:

1. Open the Host menu and click on Connect.

2. In the Connect to Host dialog box select 3270 Printer or 5250 printer as your terminal type.

3. Click the Advanced... button to display the Session Setup dialog box.

4. Select Setup Items = Printer and set the desired printer options.

5. Click OK, Click Connect...

(Return to top)

Does TN3270 Plus have TLS support?

TLs (Transport Layer Security) support is an optional, additional cost feature. You must order TN3270 Plus with the SSL feature to get TLS support.

(Return to top)

How can I limit the number of TN3270 Plus Sessions?

By default, TN3270 Plus allows 99 sessions. You can reduce the number of sessions by adding the following DWORD registry entry:

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\MaximumSessions

and setting it to the maximum number of sessions you wish to allow. Numbers from 1 to 99 are valid.

(Return to top)

How can I limit the number of instances of TN3270 Plus that users can start?

By default, TN3270 Plus allows an unlimited number of instances to be started. You can limit the number of instances by adding the the following DWORD registry entry:

HKEY_CURRENT_USER\Software\SDI\TN3270 Plus\Configuration\MaxInstances

and setting it to the maximum number of instances you wish to allow. When the maximum number of instances is reached, an attempt to start another instance results in the last started instance being brought to the foreground. If the value is set to zero, TN3270 Plus will not start. If the value is set to -1, any number of instances are allowed.

(Return to top)

9 Messages

9.1 A problem occurred while trying to create the desktop shortcut: error

Description

TN3270 Plus was unable to create the shortcut. The *error* field indicates the reason for the error.

9.2 An error occurred during the move data process -113

Description

This error comes from InstallShield which we use to package our product. This message may appear when other applications are running while you are installing TN3270 Plus. You must close all other applications prior to installing TN3270 Plus. Close all other applications and try the install again.

If closing all other applications fails, the circumvention is to unzip the TN3270 Plus self-extracting file using a zip utility. Unzip the TN3270 Plus self-extracting file to a temporary folder and then execute the setup.exe program to start the installation wizard.

If this does not work for you, please send an e-mail to support@sdisw.com. Please include your telephone number.

9.3 An error occurred during the move data process -623

Description

This error comes from InstallShield which we use to package our product. This message appears when other applications are running while you are installing TN3270 Plus. You must close all other applications prior to installing TN3270 Plus. Close all other applications and try the install again.

If this does not work for you, please send an e-mail to support@sdisw.com or fill out the Quick Contact Form. Please include your telephone number.

9.4 Are you sure you want to cancel the copy operation?

Description

You canceled a TN3270 Plus copy operation. Click the Yes button to cancel the copy operation. Click the No button to resume the copy operation.

9.5 Are you sure you want to cancel the file transfer for session n, session

Description

You pressed the Escape key requesting that an active file transfer be canceled. Click the Yes button to cancel the file transfer. Click the No button to resume the file transfer.

9.6 Are you sure you want to cancel script: filename for session n?

Description

You canceled a TN3270 Plus script. Click the Yes button to cancel the script. Click the No button to resume the script.

9.7 Are you sure you want to close session n, session name?

Description

This message warns you when you attempt to close a TN3270 Plus terminal session. You can disable this message by unchecking the "Ask for confirmation if I close a session" option in the TN3270 Plus Preferences. (Setup, Preferences)

9.8 CArchiveException thrown loading 'filename', cause: error

Description

TN3270 Plus was unable to load the specified file. The *error* field indicates the reason for the error.

9.9 Can't create toolbar

Description

TN3270 Plus was unable to create the toolbar.

9.10 Can't delete the default color scheme

Description

TN3270 Plus does not allow the default color scheme to be deleted.

9.11 Can't detete the default keypad toolbar

Description

TN3270 Plus does not allow the default keypad toolbar to be deleted.

9.12 Can't start session session_name - all sessions in use

Description

You have exceed the maximum number sessions allowed. By default TN3270 Plus allows 99 sessions, but your system administrator may have set the session limit to a lower number. Contact your system administrator.

9.13 Can't start session session_name - session nn in use

Descriptions

The specified session cannot be started because it contains a preferred session number (Setup, Host, Preferred Session Number) and that session number is already in use.

9.14 Certificate verification failed, reason: 'unable to get local issuer certficate'

Description

This message means that TLS is unable to validate the host certificate against the Root CA (Certificate Authority) certificate.

If the certificate was generated by a major Certificate Authority, the Root CA certificate should be in the Windows certificate store. Please check the TN3270 Plus "Use Windows certificate store" option (Host, Connect..., Advanced... button, Setup Items = Security, Use Windows certificate store check box.)

If the "Use Windows certificate store" option is already checked, set the "Display certificate when connected" option. (Host, Connect..., Advanced... button, Setup Items = Security, Display certificate when connected check box). Capture a screen image of the certificate and email the screen image support@sdisw.com

9.15 DDE initialization failed

Description

TN3270 Plus was unable to initialize DDE during startup.

9.16 Do you want to move your configuration files to a new location?

Description

You have changed the "Configuration File Folder" specification. Click the Yes button and your configuration files will be moved to the new location. Click the No button to leave the configuration files in the existing location. If you click the Yes button you must restart TN3270 Plus for the change to take effect.

9.17 Error 1327. Invalid Drive x

Message

Error 1327. Invalid Drive *x*

Where:

x is a drive letter.

Description

This error comes from InstallShield, the install wrapper that we use to package our software.

InstallShield verifies all Standard Windows folders are available before installing. The drive specified in the message and/or one of the standard Windows Folders on was not writeable during the install. Your drive letters may have changed and one of the Standard Windows Folders points to a drive letter than no longer exists.

Solutions

Temporary Solution

A temporary solution is to reassign the problem drive to a folder that exists.

1. Open a Windows command prompt as administrator. (Start, All Programs, Accessories, right click on Command Prompt and select run as administrator from the context menu.)
2. Type the follows command in the command prompt window and press enter:

   subst *x*: c:\

   where *x* is the drive letter specified in the error message.
3. Retry the install.

The reassignment will disappear after the computer is rebooted.

Permanent Solution

The permanent solution is to add the missing drive back to your configuration or to correct the standard windows folder registry entry that points to the missing drive and point it to a drive that exists.

The following steps show how to verify and correct the standard window folder entries. This involves modifying the registry. Modifying the registry incorrectly can damage your Windows system. Make sure you back up the registry before you make any changes.

1. Run regedit.

2. Locate the following registry key:

   HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders

3. In the right pane, look for any values that contain the drive letter specified in the error message. Right click on the value and select Modify from the context menu.
4. Change the erroneous drive letter to a valid drive letter. For example if the value is "X:\Users\Administrator\AppData\Roaming", change it to "C:\Users\Administrator\AppData\Roaming" or whatever location is appropriate for your configuration.
5. Repeat steps 3 and 4 for every incorrect value.
6. Repeat steps 3 through 5 for the following registry keys:

   KEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders
   KEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders

7. Retry the install.

9.18 Error 1722. There is a problem with the Windows Installer package

Message

Error 1722.There is a problem with this Windows Installer package. A program run as part of the setup did not finish as expected. Contact your support personnel or package vendor.

Description

This message appears when one of the TN3270 Plus install routines is blocked by an antivirus program.

This message is normally caused by:

One of the Norton Antivirus applications (e.g. Norton Security Suite or Norton 360). The issue is caused by the Norton "SONAR Advanced Protection" routine. The problem can be resolved by turning off Norton SONAR Advanced Protection while TN3270 Plus is being installed.

9.19 Error: TIMER_CLOCK SetTimer failed

Description

An error occurred when TN3270 Plus attempted to start a timer for the time of day clock.

9.20 Error copying file filename to folder folder_name

Description

An error occurred copying the specified file to the specified folder.

9.21 Error creating certificate_file

Description

An error occurred while exporting a certificate from the Windows certificate store to the TN3270 Plus certificate store. Contact SDI.

9.22 Error loading private key file 'filename': error

Description

TN3270 Plus was unable to load the specified private key file. The *error* field indicates the reason for the error.

9.23 Error reading certificate file 'filename', error

Description

TN3270 Plus was unable to read the specified certificate file. The *error* field indicates the reason for the error.

9.24 Fatal error - unable to locate SHGetSpecialFolderPath in Shell32.dll

Description

TN3270 Plus was unable to locate SHGetSpecialFolderPath. Contact SDI.

9.25 Failed to create new socket: Socket error nnnnn

Message

Failed to create new socket: Socket error *nnnnn (*WSA*xxxxxxx)*

Description

This message is caused by a Windows Socket error. Look up the Socket error found later in this chapter to find the reason for the error.

9.26 Failed to print list

Description

TN3270 Plus was unable to obtain a temporary file for print job.

9.27 filename file open failure: error

Description

The *error* field indicates the reason for the error.

9.28 File 'filename' already exists, please choose another

Description

The specified file already exists. Choose a different filename.

9.29 File Not Found

Message

drive:*path**filename.ext* file open failure: File not found

Description

The file you are attempting to access does not exist. Either click Cancel button and correct the file specification or create the missing file and click the Retry button.

9.30 INDFT017 Missing or incorrect data set name: file transfer canceled

Description

A file transfer from z/OS failed. The z/OS filename is missing or invalid. This error can be caused by any of the following:

1. The z/OS filename is not spelled correctly.
2. The z/Os file does not exist.
3. A PDS filename is specified without a member name. You can transfer a PDS member, but you cannot transfer an entire PDS.

.

9.31 Invalid Configuration Files Folder!

Description

You have specified an invalid configuration files folder. Open the TN3270 Plus Preferences dialog (Setup, Preferences...) and enter a valid configuration files folder. You must restart TN3270 Plus for the configuration files folder change to take effect.

9.32 Invalid margin specification

Description

The specified margin is not valid.

9.33 Invalid port number in host name list

Description

The port number is greater than 65535.

9.34 Keyboard-interactive authentication type is not valid for SSHv1

Description

The Keyboard-interactive authentication type is not valid for SSHv1. Choose Password or Private key authentication or change the Encryption Protocol to SSHv2.

9.35 No files to copy

Description

TN3270 Plus could not find any configuration files to copy.

9.36 One or more files to be copies already exist in the target folder

Description

One or more of the TN3270 Plus configuration files already exists in the target folder. Click the Yes button to overwrite the existing files. Click the No button to cancel the copy operation.

9.37 Please enter a Proxy Name or IP address

Description

Enter a Proxy Name or IP address.

9.38 Please enter a name for this session

Description

Enter a name for the session.

9.39 Please enter a name for this shortcut

Description

You clicked the OK button in the **Create Desktop Shortcut** dialog without entering a name. You must enter a shortcut name to create the shortcut.

9.40 Please enter a value for prompt

Description

A value is required for the specified prompt.

9.41 Please select an authentication type

Description

You must specify an SSH authentication type for your session. (Setup, Security, Authentication Type)

9.42 Please select a color scheme

Description

You must specify a color scheme for the session.

9.43 Please select a keypad toolbar

Description

You must specify a keypad toolbar for the session.

9.44 Please select a terminal type

Description

You must specify a terminal type for the session.

9.45 Please type the name of the private key file

Description

A private key file is required for Public key authentication. Contact the network administrator at the host computer site to get your private key file.

9.46 Private key check failure, error

Description

TN3270 Plus was unable to load the private key file. The *error* field indicates the reason for the error.

9.47 Private key passphrase

Description

Enter the private key passphrase.

9.48 Re-sending PC file

Description

TN3270 Plus file transfer is resending the specified block.

9.49 Save change to name

Description

Click the OK button to save your changes.

9.50 Save changes to name color scheme

Description

Click the OK button to save the changes to you color scheme.

9.51 Socket error 0 (WSAEUNDEFINED)

Message

Host www.xxx.yyy.zzz port nnn Is Unreachable: Socket error 0 (WSAEUNDEFINED)

Description

You are attempting to connect to an Internet host address and your computer is not connected to the Internet.

9.52 Socket error 10022 (WSAEINVAL)

Message

Host *hostname* port *nnn* Is Unreachable: Socket error 10022: WSAEINVAL

Description

This error is normally caused by one of the following:

1. The hostname specified in the Host Name edit box of the Connect to Host dialog box (Host, Connect...) is not valid.

   The Host Name should be the dotted IP address or the DNS name of the host computer. Every computer on the Internet has an IP address that looks something like this "128.228.1.20". To make this address easier to remember, the IP address may be registered and assigned a DNS name. For example, acmecomputer.com. When someone tries to connect to acmecomputer.com a DNS server on the Internet translates that to 128.228.1.20. In this example, either acmecomputer.com or 128.228.1.20 could be used as the host name.

   Think of the host name as the address of the host computer. It tells TN3270 Plus where the host computer is. You need to find out what the host name or IP address is for the computer you want to connect to. Try talking to the network administrator at the host computer site if you are unsure.

2. On an intranet (internal network), you may need to make an entry in the Windows "hosts" file to equate the hostname to the IP address.

   You will find the Windows hosts file here:

Windows 11/10/ 8/ 7/XP Window Server 2022/2019/2016/2012/2008/2003	c:\windows\system32\drivers\etc\hosts
Windows 2000	c:\winnt\system32\drivers\etc\hosts

   This file should contain an entry that looks something like this.

   192.168.1.5 *hostname*

   Where:

   hostname is the hostname from the error message.
   192.168.1.5 is the IP address of the host on your network.

9.53 Socket error 10051 (WSANETUNREACH)

Message

Session x (hostname:port) Socket Error 10051 (WSAENETUNREACH).

Description

A socket operation was attempted to an unreachable socket.

This usually means TN3270 Plus cannot find a route to reach the remote host.

This error is normally caused by an incorrect IP address in the "Host Name" edit box in the "Connect to Host" dialog box.

Try pinging the host computer to see if it is responding. You can this from the Windows command prompt (Start, All Programs, Accessories, Command Prompt) by entering "ping hostname" or "ping IP address" on the command line. If Ping fails, talk your network administrator and get the correct IP address.

9.54 Socket error 10060 (WSAETIMEDOUT)

Message

Session x (*hostname*:*port*) Socket Error 10060 (WSAETIMEDOUT). A connection attempt failed because the connected party did not respond properly after a period of time, or established connection failed because connected host has failed to respond.

Description

This message means that TN3270 Plus has sent communication to the host computer and the host computer did not respond before the end of the timeout period. The timeout period is monitored by the Windows socket, when the end of the timeout period is reached the Windows socket indicates the error to TN3270 Plus and TN3270 Plus reports the error.

This message is normally caused by one of the following:

1. An incorrect IP address or Host Name in the "Connect to Host" dialog box.

2. An incorrect port number specified in the "Connect to Host" dialog box.

3. A proxy server or firewall is blocking communication between TN3270 Plus and the host computer.

4. Your network uses a VPN (Virtual Private Network) and the VPN is not connected.

Try pinging the host computer to see if it is responding. You can this from the Windows command prompt (Start, All Programs, Accessories, Command Prompt) by entering "ping hostname" or "ping IP address" on the command line. If Ping times out, there is a problem at the host. Talk to a network administrator at the host site and see if they can help you.

9.55 Socket error 10061 (WSAECONREFUSED)

Message

Session x *(hostname*:*port*) Socket Error 10061 (WSAECONNREFUSED).

Description

No connection could be made because the target machine actively refused it.

This message is normally caused by one of the following:

1. The remote system rejected your attempt to connect with it, either because no server is listening on the specified port or it's unable to accept any additional connections. Verify you are using the correct port number and then contact the Network Administrator at the host computer site.

2. You have specified "localhost" as the Host Name in the Connect to Host dialog box. "localhost" is the host name of your PC. Replace localhost with the Host Name of the computer you wish to connect to.

   The Host Name should be the dotted IP address or the DNS name of the host computer. Every computer on the Internet has an IP address that looks something like this "128.228.1.20". To make this address easier to remember, the IP address may be registered and assigned a DNS name. For example, acmecomputer.com. When someone tries to connect to acmecomputer.com a DNS server on the Internet translates that to 128.228.1.20. In this example, either acmecomputer.com or 128.228.1.20 could be used as the host name. Think of the host name as the address of the host computer. It tells TN3270 Plus where the host computer is. You need to find out what the host name or IP address is for the computer you want to connect to. Try talking to the network administrator at the host computer site if you are unsure.

9.56 Socket error 11001 (WSAHOST_NOT_FOUND)

Message

Failed to create new socket: Socket error 11001 (WSAHOST_NOT_FOUND). No such host is known.

Description

The name is not an official host name or alias, or it cannot be found in the database(s) being queried. This error may also be returned for protocol and service queries, and means that the specified name could not be found in the relevant database.

This message is normally caused by one of the following:

1. Specifying an incorrect host name in the "Host Name" edit box in the Connect to Host dialog box.
2. Specifying the incorrect IP version level in the Connect to Host dialog box. For example, specifying IPv6 for an IPv4 connection.

9.57 Socket error 11004 (WSANO_DATA)

Message

Failed to create new socket: Socket error 11004 (WSANO_DATA). The requested name is valid, but not data of the requested type was found.

Description

The requested host name is valid and was found in the database, but the associated data is not correct.

This message is normally caused by one of the following:

1. Specifying an incorrect host name in the "Host Name" edit box in the Connect to Host dialog box.
2. Specifying the incorrect IP version level in the Connect to Host dialog box. For example, specifying IPv6 for an IPv4 connection.

9.58 SSLv3 handshake failure

Message

SSLv3 handshake failure: Socket error *n* (*ssl_error_code*)

Description

This message indicates and error attempting to make an SSLv3 connection. The following table provides the meaning of the different SSL error code.

Code	Meaning

SSL_ERROR_SYSCALL	An O/O error occurred.
SSL_ERROR_SSL	This is usually an error in the TN3270 Plus SSL protocol.

9.59 That name already exists, please choose another

Description

The name you have specified already exists. Use a different name.

9.60 The license code is not valid for this release

Description

The license code you entered is for an older release of TN3270 Plus. The license code is generated for a specific release and will not function with newer releases of TN3270 Plus.

The product release level is three digits separated by periods, for example, 3.4.0. The license code only checks the first two digits, so a license code created for 3.4.0 will work with releases 3.4.0 through 3.4.9. In addition, license codes for release 3.3 and above will work with the next higher release. For example, a release 3.4 license code will also work with release 3.5 (3.5.0 - 3.5.9). However, if you try to use a release 3.4 license code with release 3.6.0 or higher you will get the above error message.

If you have purchased a maintenance and support subscription, updates are free. Send an e-mail to sales@sdisw.com with your current "licensed to" name and the new release level and you will be e-mailed a license code for the new release.

If you do not have a maintenance and support subscription, you will need to subscribe to our maintenance and support plan in order to receive the new product release.

9.61 The 'Licensed To' field is incorrect

Description

The name in the "Licensed to" name does not match the name encrypted into the license code.

1. The "Licensed to" name is incorrect or contains a typographical error.
2. The "Licensed to" name is case sensitive make sure all letters are in the correct case.
3. Make sure the "Licensed to" name has the correct number of spaces.
4. The "Licensed to" name and license code are issued in pairs. The "Licensed to" name must correspond to the license code you received.

9.62 The 'License Code' field is incorrect

Description

The "License code" field contains a typographical error. Make sure that all the characters are correct.

9.63 The name you have chosen contains an invalid character

Description

You specified a shortcut name with an invalid character. The following characters are invalid: \ / : * ? " < > |

Remove the invalid character.

9.64 The servers RSA key fingerprint is: fingerprint

Message

The servers RSA key fingerprint is: *fingerprint.* If you trust host click Yes to save the fingerprint, click No to connect without saving it, or click Cancel to abort the connection.

Description

Self explanatory.

9.65 There are still n sessions active

Message

There are still *n* sessions active. Are you sure you want to exit?

Description

This message warns you if you attempt to close TN3270 Plus while there are still one or more terminal sessions active. You can disable this message by unchecking the "Ask for confirmation before terminating sessions are still connected" option in the TN3270 Plus Preferences. (Setup, Preferences)

9.66 This is an evaluation version of TN3270 Plus

Message

This is an evaluation version of TN3270 Plus.

THE EVALUATION PERIOD HAS *NN* DAYS LEFT.

To purchase, click the Registration Form....

Description

This message is appears when your are running the TN3270 Plus evaluation version. The message indicates how many more days the evaluation version will run before it expires. If you are planning to purchase a TN3270 Plus license, you should place your order one week before the product expires to insure your permanent license code is delivered on time. To place an order, click on the Purchase Online or Registration form button. To continue your evaluation of TN3270 Plus click the Continue button to close the dialog box. When you receive your permanent license code, you can click the License Code button to enter your license code.

9.67 TLSvn.n handshake failure

Message

TLSv*n.n* handshake failure: *error_descripton*

Description

This message indicates and error attempting to make a TLS connection. *n.n* is the TLS version level. The following table provides the meaning of the different error descriptions.

Code	Meaning
wrong version number	The host computer does not support the requested TLS version.

9.68 To return to normal window

Message

To return to normal window mode, activate the pop-up menu by pressing the Alt+SpaceBar key, then select \"View, Full Screen\" again.

Full screen mode removes the window frame, toolbar and status bar, adds the Operator Information Area (OIA) and maximizes the terminal window.

You may continue to use your session settings for the toolbar, status bar and OIA by selecting that option below.

Description

This informational message appears when you switch TN3270 Plus to full screen mode.

9.69 TN3270 Plus was not installed correctly, please re-install

Description

The TN3270 Plus install did not complete successfully. Uninstall and reinstall and make sure you have administrative privileges during the install.

9.70 TRANS13 Error writing file to host

Message

TRANS13 Error writing file to host: file transfer canceled

Description

The host program IND$FILE detected an error during a receive operation. You will receive this error if you are using WSF mode and the host operation system does not support it. Try changing from WSF mode to Screen Images mode. (Host, File Transfer, File Transfer Mode = Screen Images)

9.71 TRANS17 Missing or incorrect TSO data set name

Message

TRANS17 Missing or incorrect TSO data set name: file transfer canceled

Description

And incorrect data set name was specified. This error can also occur if a TSO data set name is not enclosed in single quotation marks.

9.72 TRANS18 Incorrect option specified

Message

TRANS18 Incorrect option specified: file transfer canceled

Description

An incorrect file transfer option was specified. This can occur if an incorrect blksize or lrecl is specified. If you

receive this error, try specifying 0 (zero) for the blksize and lrecl.

9.73 Unable to access the SecureBlackBox Dll's

Message

Unable to access the SecureBlackBox Dll's

SSH support will be inoperative.

Description

TN3270 Plus was unable to access the SecureBlackBox Dll's required for SSH support. This error could be caused by one of the following:

1. One or more of the SecureBlackBox Dll's were not installed.

2. One or more of the SecureBlackBox Dll's were deleted.

3. One or more of the SecureBlackBox Dll's were not registered.

The SecureBlackBox Dll's are: BaseBBox.dll and SSHBBoxCLi.dll

To register the there DLL's manually use theregsvr32.exe. Administrative privileges are required. For example:

```
regsvr32.exe  "c:\Program files (x86)\SDI\TN3270 Plus\baseBBox8.dll"
regsvr32.exe  "c:\Program files (x86)\SDI\TN3270 Plus\SSHBBoxCLi.dll"
```

9.74 Unable to create folder 'folder'': error

Description

TN3270 Plus was unable to create the specified folder. The *error* field indicates the reason for the error.

9.75 Unable to launch the default web browser

Description

TN3270 Plus was unable to launch your default web browser.

9.76 Unable to open debug trace file. Continue anyway?

Description

TN3270 Plus was unable to open the debug file. Click the Yes button to continue TN3270 Plus startup without debugging active. Click the No button to terminate TN3270 Plus startup. To resolve the issue specify the debug file in a folder that has write access.

9.77 Unable to open trace file 'filename': error

Description

TN3270 Plus was unable to open the trace file. The *error* field indicates the reason for the error.

9.78 Unable to save settings to filename: error

Message

Unable to save settings to *filename: error.*

WARNING! Windows may not allow you to save changes to your session setting in the existing location *(filename*). You should move all you configuration files to the preferred location: *filename*

Description

You do not have access to the specified folder for the reason specified. The *error* field indicates the reason for the error. You can either change the permissions for the folder to allow access or copy your TN3270 Plus configuration files to a location where you have access. SDI recommends you move your files to the default location.

9.79 WARNING! Vista may not allow you to save changes to your sesson settings in the existing location

Message

WARNING! Vista may not allow you to save changes to your session settings in the existing location (*'folder'*). You should move all your configuration files to the preferred location: *'folder'*

Description

Move the TN3270 Plus configuration file to the preferred location specified in the message.

9.80 WinHLLAPI Session A in use elsewhere

Description

The specified WinHLLAPI session ID is already in use by another active TN3270 Plus session. You can resolve this problem by doing one of the following:

1. Close the session that is currently using the specified WinHLLAPI session ID.
2. Change the session you are attempting to connect to a different WinHLLAPI session ID. (Host, Connect, Advanced button, Setup Items = Host, WinHLLAPI Session ID = B)
3. Remove the WinHLLAPI session ID specification from the session you are attempting to connect. (Host, Connect, Advanced button, Setup Items = Host, WinHLLAPI Session ID = None)

9.81 You must disconnect this session before changing the terminal type

Description

You cannot change the terminal type while a session is active. To change the terminal type, you must disconnect the session (Host, Close), open the Setup menu, select Sessions..., Setup Items = Terminal and then change the selection in the Terminal Type list box.

9.82 3270 Attn key string 'key_string' is invalid - using default (default_key_string)

Description

The 3270 Attn key string specification is invalid. See the Changing the 3270 Attention and System Request Keys topic for information about changing the 3270 Attn key specification.

9.83 3270 Sysreq key string 'key_string' is invalid - using default (default_key_string)

Description

The 3270 Sysreq key string specification is invalid. See the Changing the 3270 Attention and System Request Keys topic for information about changing the 3270 Sysreq key specification.

10 Glossary

10.1 Glossary

AID
command prompt
control file
current session
daemon
DDE (Dynamic Data Exchange)
domain name
EAB (Extended Attribute Byte)
evaluation version
fixed-pitch font
host computer
Hotspot
intranet
IP address
Operator Information Area (OIA)
LPD
LPR
ping
POP3
port number
resource name
system command
SSH (Secure Shell)
SSL (Secure Socket Layer)
TLS (Transport Security Layer)
TSO (Time Sharing Option)

VM (Virtual Machine)
Windows Temp folder

Index

- 5 -

- 7 -

- 8 -

- A -

- B -

- C -

- D -

- E -

- F -

- G -

- H -

- I -

- J -

- K -

- L -

- M -

- N -

- O -

- P -

- Q -

- R -

- S -

- T -

- U -

- V -

- W -

- X -

- Y -

Back Cover