IBM

Configuration Program User's Guide for Nways Multiprotocol and Access Services Products

IBM

Configuration Program User's Guide for Nways Multiprotocol and Access Services Products

Notices

References in this publication to IBM products, programs, or service do not imply that IBM intends to make them available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

> IBM Director of Licensing
> IBM Corporation
> North Castle Drive
> Armonk, NY 10504-1785
> U.S.A.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX	Nways	RS/6000
AIX/6000	Operating System/2	SystemView
NetView	OS/2	

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

Intel and Pentium are trademarks of Intel Corporation in the United States and/or other countries.

Other company, product, and service names may be trademarks or service marks of others.

About This Manual

This manual explains how to install and use the Configuration Program on various platforms to configure an Nways device for your network.

This manual assumes a knowledge of networking concepts.

Who Should Read This Manual

The intended user of this book is the person who will configure an Nways device using the configuration program.

How This Manual Is Organized

This manual is divided into the following chapters:

- "Chapter 1. Installing the Configuration Program" on page 1, which describes the installation requirements for the IBM Nways Access Integration Services, Multiprotocol Access Services, Multiprotocol Routing Services, Multiprotocol Switched Services, and Multiprotocol Switched Services Clients Configuration Program and how to install the configuration program.

- "Chapter 2. Starting the Configuration Program" on page 5, which describes how to start the configuration program and some operational considerations.

- "Chapter 3. Using the Configuration Program" on page 9, which describes how to use the configuration program, some of its navigational features, and an overview of the configuration process.

Contents

Features of the Configuration Program

The Configuration Program allows you use a point and click interface to create configurations for the various models of devices. Some of the features of the configuration program that make configuring a device easier than using the command line interface are:

- You can save a configuration for later completion.
- You can retrieve an existing configuration from a device and either update it or use it as a template for configuring another device.
- You can use any configuration in the database as a template for a different device.
- You can configure the protocols in any order unless there are inter-protocol dependencies.
- You can group configurations together in configuration databases that reflect your network topology.
- You can upgrade configurations to the current format. For example, configurations that are stored in a configuration database that was created with previous versions of the configuration program can be upgraded.
- You can verify the completeness of a configuration before loading that configuration to a device.
- You can use the guided configuration. It describes the procedures to follow when you configure protocols.
- You can use the context-sensitive helps for panels and parameters to get more information as needed.
- You can save configurations as backups or store multiple versions for the same router.
- You can use the command line facility to automate many tasks including reading, writing, sending, and retrieving configurations.
- You can create, edit, and read ASCII-formatted configuration files.

Library Ordering Information

Obtaining Softcopy Information

Softcopy BookManager READ library information is available for many of the IBM 2210, IBM 2212, IBM 2216, and IBM 8210 publications in the *IBM Networking Systems Softcopy Collection Kit.* To place a single order for the CD-ROM, use form number SK2T-6012. To place a single order for the 3480 cartridge, use form number SK2T-6013.

Yearly subscriptions to the *IBM Networking Systems Softcopy Collection Kit*, product number 5636-PUB, are available through your branch office representative. Order feature code 2003 and media code 5003 for CD-ROM format. Order feature code 2004 and media code 5004 for 3480 cartridge format.

To order the CD-ROM that contains the 2210-related publications, use form number SK2T-0404.

To order the CD-ROM that contains the 2212-related publications, use form number SK2T-0435

To order the CD-ROM that contains the 2216- and Network Utility-related publications, use form number SK2T-0405.

To order the CD-ROM that contains the 8210-related publications, use form number SK2T-0425.

To order the CD-ROM that contains the MSS Client-related publications, use form number SK2T-0429.

System Library Subscription Service

The IBM 2210 publications are available via the System Library Subscription Service (SLSS). Use machine type 2210 and program number 5801-ARR to receive 2210 publications.

The IBM 2212 publications are available via the System Library Subscription Service (SLSS). Use machine type 2212 and program number 5639-F73 to receive 2212 publications.

The IBM 2216 publications are available via the System Library Subscription Service (SLSS). Use machine type 2216 and program number 5765-D47 to receive 2216 publications.

The Network Utility documentation is included in the IBM 2216 publications.

The IBM 8210 publications are available via the System Library Subscription Service (SLSS). Use machine type 8210 and program number 5765-368 to receive 8210 publications.

Summary of Changes to Fifth Edition

Changes to This Manual

- Program requirements have changed.
- Installation procedures have changed.
- Migration procedure has changed.
- Added information for the syntax of the **config** command.

Technical changes and additions are indicated by a vertical line (|) to the left of the change.

Changes to the Configuration Program Since the Fourth Edition

The following changes were made to the configuration program since the last edition of this manual:

Requirements
The hardware and software requirements for the configuration program have changed.

Fonts The configuration program no longer supports changing fonts. However, you can use the various operating system functions to make font changes.

-logMessages Command Replaced by -storeMessages
The command arguments remain the same.

List Item Selection
The configuration program now supports selecting multiple items from selection lists. The selection mechanism is the standard mechanism for the operating system.

Chapter 1. Installing the Configuration Program

This chapter describes the minimum requirements and the procedure for installing and starting the IBM Nways Access Integration Services, Multiprotocol Access Services, Multiprotocol Routing Services, Multiprotocol Switched Services, and Multiprotocol Switched Services Clients Configuration Program.

Important: Before proceeding, read the README file to obtain the latest information about requirements and any changes to the installation procedures.

Hardware and Software Requirements for the Configuration Program

Table 1 lists the minimum requirements for the operating systems supported by the configuration program.

Table 1. Configuration Program Requirements for Supported Operating Systems

Requirement	IBM AIX	IBM Operating System/2	Microsoft Windows 95 or Windows NT
Workstation	RS/6000 Model 250/410 or higher with minimum 80-Mhz CPU	IBM-compatible PC with Intel Pentium 166-MHz or higher processor (see Note 1)	IBM-compatible PC with Intel Pentium 166-MHz or higher processor (see Note 1)
RAM (minimum)	64 MB	48 MB	48 MB
CD-ROM	yes	yes	yes
Free Disk Space	54 MB	37 MB	54 MB
	Additionally, you will need 1 to 3 MB for each configuration file.	Additionally, you will need 1 to 3 MB for each configuration file.	Additionally, you will need 1 to 3 MB for each configuration file.
Swapper Size	N/A	10 MB	N/A
Display (minimum)	Graphics (1024 x 768 x 256 colors)	SVGA (1024 x 768 x 256 colors)	SVGA (1024 x 768 x 256 colors)
Mouse	3-button	2-button (see Note 2)	2-button (see Note 2)
Software	AIX for RS/6000 V4.1.5 or higher AIXWindows Environment/6000	OS/2 3.0 or higher IBM TCP/IP 1.2.1 for OS/2 or higher	Microsoft Windows 95 or Windows NT

Notes:

1. More memory and faster processors improve the program's performance.
2. Functionally equivalent to a 3-button mouse.
3. You must make sure that TCP/IP is *installed* and *running* before using the "Single Send or Retrieve" or "Multiple Send or Retrieve" functions. On MS Windows platforms, the configuration program will hang if TCP/IP is not installed and running.

The AIX version of the conversion program requires special fonts. Note that the 2212 Access Utility does not use the conversion program. For this program to run

1

correctly, you must have **/usr/lib/X11/fonts/75dpi/** in your list of directories. If you do not have this directory in your Xstation Font Path, at your Xstation, enter:

> **xset fp+ /usr/lib/X11/fonts/75dpi/**

This command adds the conversion program font directory to your Xstation Font Path. Place the above line in your .profile for future use.

Obtaining the Configuration Program Electronically

You can obtain the configuration program electronically, from the IBM Networking Technical Support Home Page. Follow this procedure to download the configuration program:

1. Access the IBM Networking Home Page at:

   > **http://www.networking.ibm.com**

2. Click **Tech Support** to access the IBM Networking Technical Support page.
3. Select the product number (for example, 2210, 2212, 2216, Network Utility, or 8210) or the product name.
4. Click **Downloads** to access the Code Downloads page. You will be linked to a page at:

   > **http://www.networking.ibm.com/support/code.nsf/***product***code**

   where *product* is the product number or name. For example, *product***code** for the 2210 is **2210code**, and *product***code** for the Network Utility is **networkutilitycode**

   The Code Downloads page for the 2210 is at:

   > **http://www.networking.ibm.com/support/code.nsf/2210code**

   The Code Downloads page for the Network Utility is at:

   > **http://www.networking.ibm.com/support/code.nsf/networkutilitycode**

5. Select the desired code level for the configuration program that you want to download. On the next panel, in the **Files in this Package** box at the bottom of the panel, download the file you want into a temporary directory. (Consult the README file to determine which version of the configuration program and which file you need, and ensure that you have enough space on the hard drive.)
6. To install the configuration program, continue at "Installing the Configuration Program from the IBM Networking Home Page".

Installing the Configuration Program from the IBM Networking Home Page

The procedure you use for installing the configuration program depends on the operating system under which the program will run. Before you proceed, make sure that you have write access to the directory in which you are installing it. The installation process writes program files to the directory during installation and the configuration program writes information to the directory while it is running. See the appropriate README file for more detailed information.

- To install the configuration program for AIX, go to step 8 on page 3 in "Installing on AIX" on page 3.

- To install the configuration program for OS/2, enter **os2inst** on an OS/2 command window and follow the prompts that appear on the panel to complete the installation.
- To install the configuration program for Windows, run the self-extracting **.exe** file and follow the prompts that appear on the panel to complete the installation.

Installing the Configuration Program from the CD-ROM

The procedure you use for installing the configuration program depends on the operating system under which the program will run. "Installing on AIX", "Installing on OS/2" on page 4, and "Installing on Windows" on page 4 describe the steps you take to install the configuration program, based on your operating system.

Before installing the configuration program, make sure that you have write access to the directory in which you are installing the configuration program. The installation process writes program files to the directory during installation and the configuration program writes information to the directory while it is running.

Installing on AIX

To install the configuration program on AIX:

1. Ensure that you are logged on as the "root" user. (The **whoami** command will show root at the command prompt.) If not, enter **su root**.
2. Insert the appropriate Configuration Program CD-ROM into the CD-ROM drive.
3. Create a directory named cdrom, if one does not exist:

    mkdir /cdrom
4. Mount the CD-ROM:

    mount -v cdrfs -r" /dev/cd0 /cdrom
5. Make a directory into which you will copy the installation files.
6. Change to the directory you created.
7. Copy the INSTALLP file, *config/aix/pgmname* from the CD-ROM drive to your directory, where *pgmname* is the name of the INSTALLP file that is appropriate to your load of the configuration program, as documented in the README file. For example:

    cp */cdrom/config/aix/pgmname*

    where */cdrom/* is the mount point for your CD-ROM.
8. Run SMIT and do the following:

    Note: The following steps are based on AIX 4.1.5. The steps are slightly different if you are using AIX 4.2 or higher.

    a. Select **Software Installation and Maintenance**.
    b. Select **Install and Update Software**.
    c. Select **Install/Update Selectable Software (Custom Install)**.
    d. Select **Install Software Products at Latest Level**.
    e. Select **Install New Software Products at Latest Level**.
    f. Enter the fully qualified file name of the INSTALLP file from step 7.
    g. Select the installation program.
    h. Select **OK**.

| i. Select **OK**.

| j. Select **Done**.

| k. Select **Cancel**.

| 9. Remove the directory you created in step 5.

| 10. Change directory to **/usr/lpp/***pgmname*.

| 11. Unmount the CD-ROM:

| **unmount /cdrom**

| 12. Enter **exit** to end the root user session.

Installing on OS/2

To install the configuration program on OS/2:

1. Open an OS/2 command window.

2. Insert the appropriate CD-ROM into the CD-ROM drive.

3. Change directory to **/config**.

4. Enter **os2inst** and follow the prompts that appear on the panel to complete the installation.

Installing on Windows

To install the configuration program on Windows:

1. Insert the appropriate CD-ROM into the CD-ROM drive.

2. If autoread is on, the installation program starts and displays an installation dialog. Otherwise, change directory to **/config/win32** and run setup on the CD-ROM.

3. Follow the prompts that appear on the panel to complete the installation

Chapter 2. Starting the Configuration Program

This chapter describes:
- The procedures for starting the program
- Using configuration files created by earlier versions of the program

Starting the Configuration Program

To start the configuration program on AIX, see "Starting the Configuration Program on AIX".

To start the configuration program with AIX Netview/6000, see "Starting the Configuration Program with AIX NetView/6000".

To start the configuration program on OS/2 or Windows see "Starting the Configuration Program on OS/2 or Windows" on page 6.

Starting the Configuration Program on AIX

Before starting the configuration program, X-Windows must be running on the workstation.

To start the configuration program:
1. Enter **xinit** to start X-Windows, if not already started.
2. Change directories to the directory that contains the configuration program, for example:

   cd /usr/lpp/masv3r2
3. Enter **config**.

See "Using the Command Line Facility" on page 16 for details about using the command line facility.

Starting the Configuration Program with AIX NetView/6000

If you have NetView/6000 you can start the configuration program directly from the menu bar. If the network contains routers of different types (for example, 2210 and 6611) or configuration programs of different versions and releases, you can start any configuration program installed on the machine directly from the menu bar.

To use multiple configuration programs, make sure that each configuration program is in a separate subdirectory from the other configuration programs and NetView/6000.

For each configuration program that you want to start from the menu bar:
1. Create a file in the /usr/OV/registration/C subdirectory that contains the code for the menu bar selection item.
2. Change the command statement in the file to point to the path of the desired configuration program.
3. Save the file with a name that indicates the type, version, and release of the configuration program you are starting.

Figure 1 is an example of the code contained in a file named MRSV32 that points to the configuration program in directory /usr/epp/mrsv32. The **_C** in the line under MenuBar "Tools" is the accelerator character (hot key) for the menu item. This character must be in the character string directly to the left of the action, and cannot be duplicated in another menu item.

For more information, such as how to create a sub-menu that contains all of the configuration program menu items, refer to the *AIX SystemView NetView/6000 V2R1 Programmer's Guide,* SC31-7022.

```
/*
**  Global Tools
*/
Application "2210 MRS Configuration V3R2"
{
    Description {
         "2210 MRS Configuration V3R2"
    }
    /*
    **  Tools
    */
    MenuBar "Tools"
    {
       "2210 MRS Config Tool V3.2" _C f.action "cfgtool";
    }
    Action "cfgtool"
    {
        Command "/mrs/config/rel320/config";
    }
}
```

Figure 1. Netview/6000 Menu File MRSV1R1

Starting the Configuration Program on OS/2 or Windows

To start the configuration program under OS/2:
1. Locate and open the configuration program folder.
2. Double-click the Configuration Program icon to start the program. The title window for the configuration program displays.

To start the configuration program under Windows, select **Start→Programs→***folder***→***pgmname* where *folder* and *pgmname* are:

Folder	Pgmname
Access Integration Services	aisv32
Multiprotocol Access Services	masv32
Multiprotocol Routing Services	mrsv32
Multiprotocol Switched Services	mssv21
Multiprotocol Switched Services Family Clients	msscv21
Network Utility	netutilv32

After the Configuration Program Starts

When the program starts, it uses the options defined in the **config.ini** file. If the file does not exist in the configuration program's directory, the configuration program uses default options and creates a new **config.ini** file. To change the options, select **Options** on the configuration program menu bar. See "Changing the

Upgrading/Migrating Previous-Level Configuration Files

This version of the configuration program cannot use configuration database files with **.cdb** file extensions (created by earlier versions of the configuration program). You must upgrade to the new version format (**.csf** file extension). If the only copy of the existing configuration is in the router or in a binary format (**.cfg**) file, before starting the migration process you must use the version of the configuration program that created the existing configuration to create configuration database (**.cdb**) files, as described at "Creating a Configuration Database File", and then convert the files to **.csf** format as described at "Converting Configuration Database Files to Configuration Storage Format Files".

If your configuration database files were created by an earlier version of the configuration program, but already have **.csf** file extensions, go to "Migrating a CSF-Format Configuration File to the Latest Release" on page 8

Creating a Configuration Database File

To create a configuration database file using a previous version of the configuration program:

1. Start the previous version of the configuration program.
2. Select **Configure** → **Read router configuration** to read the existing configuration into the configuration program.

   Another way to read an existing configuration into the program is to select **Configure** → **Communications** → **Single** →**Retrieve**.
3. Select either **Configure** → **Save configuration** or **Save configuration as**.
4. Save the configuration into a file with a **.cdb** file extension.
5. Continue with "Converting Configuration Database Files to Configuration Storage Format Files" to convert the **.cdb** file into a **.csf** file.

Converting Configuration Database Files to Configuration Storage Format Files

All configuration database files (**.cdb**) from prior versions of the configuration program must be converted to the configuration storage format (**.csf**). You can use the conversion program to do this. It provides a dialog in which you specify which configuration database files you want to convert. Follow these steps to use the conversion program:

1. Start the Conversion Program.

   AIX

      a. Change to the directory that contains the configuration program.
      b. Enter **converter**.

         Note: See "Hardware and Software Requirements for the Configuration Program" on page 1 for AIX font requirements for the Conversion Program.

   OS/2

      a. Locate and open the configuration program folder.
      b. Select the **Conversion Tool** icon to start the conversion program.

Windows 95 or Windows NT

    a. Locate the configuration tool program from **Start**.

    b. Select **Conversion Tool**.

2. Enter the directory that contains the database file in the field, *Directory containing CDB.*

3. Using the radio buttons, select whether you want to convert all configurations in the configuration database or selected configurations. If you are converting only selected configurations, continue with step 4, otherwise go to step 5.

4. Select the configurations you want to convert from the Convert Configs list.

5. Specify the directory that will contain the new configuration files (**.csf** format).

6. Select **Convert**. The program then converts the specified configurations into separate configuration files in the Configuration Storage Directory (CSD).

7. Follow the procedure in "Migrating a CSF-Format Configuration File to the Latest Release" to upgrade these new **.csf** files for compatibility with the latest release of the configuration program.

Migrating a CSF-Format Configuration File to the Latest Release

To convert a configuration file that is already in **.csf** format for use by the current configuration program:

1. Start the latest version of the configuration program.

2. Select **File** → **Open configuration**. Select the configuration from the Available configurations list.

3. Select **Open**. The configuration program asks if you want to upgrade the configuration to the current version.

4. Select **Configure** → **Save configuration as** to save the configuration to a new file with a new name. Saving the configuration with a new name allows you to keep a copy of the existing configuration file for possible future use.

Chapter 3. Using the Configuration Program

This chapter describes how to use the configuration program and explains its navigation features.

Before Using the Configuration Program

Before using the configuration program to create configurations for your router please read the README file that is contained in the directory in which the configuration program is installed or on the CD-ROM. The README file contains the most current directions for using the configuration program.

Important: The configuration program that you use must match the software that resides on the Nways device that you are configuring. Ensure that you have the latest software level by comparing the software you received with the files available at the web sites named in "Obtaining the Configuration Program Electronically" on page 2.

Understanding the Navigation and Configuration Windows

After you start the configuration program and clear the copyright notice, the program displays two windows similar to those in Figure 2 on page 10 and Figure 3 on page 12.

The narrower window (on the left) is the *Navigation Window* and the wider window (on the right) is the *Configuration Window.*

Configure the router by selecting items in the Navigation Window and then providing the details for each selected item in the Configuration Window. When you have completed entering the configuration details, select a new item to configure by selecting the item in the Navigation Window.
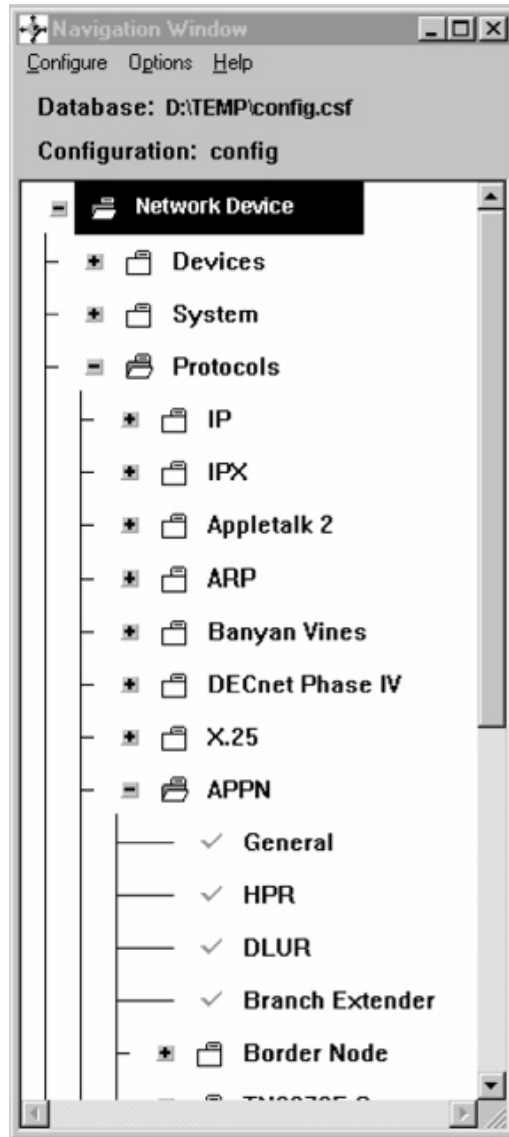
Figure 2. The Navigation Window

Using The Navigation Window

The Navigation Window contains:

- A title bar
- A menu bar
- A configuration description area
- A scrollable list of features and protocols that you can configure

This manual refers to the scrollable list as a navigation list. As you select items in the navigation list, the program displays a configuration panel in the Configuration Window. If you select a folder item, the configuration program displays help for configuring the feature that is represented by the folder in the Configuration Window. A folder is an item in the Navigation Window that has a folder icon next to it. You can copy the help text from the Configuration Window by pressing F1 at this time.

If there is an error or if you have not supplied required information in the current configuration, a question mark appears next to that item in the navigation list.

If there are no errors, a check mark (✔) appears next to that item in the navigation list.

The title bar of the Configuration Window displays the identity of the item in the navigation that you have selected.

The Navigation Window's menu bar contains the following options:

Configure

Options

Help

The area below the menu bar is the configuration description area. It displays the following information:

- The path and name that contains the configuration. The default path is the directory in which you installed the configuration program, and the default filename is **config.csf**.
- Filename of the current configuration (the default is **config**)
- The router model and, if applicable, adapter type you are configuring (the default is set in the Options menu)

Another feature of the Navigation Window is a pop-up menu that you invoke by pressing mouse button 2 (middle mouse button or the right mouse button on a 2-button mouse) on any selected folder in the navigation list. The pop-up menu allows you to:

- Fully expand the navigation list.
- Expand a portion of the navigation list out of a folder.
- Display a history pop-up menu of configuration items you have selected. From the history pop-up you can access any of the previously selected configuration items. On OS/2, Windows 3.x, and AIX, press and hold mouse button 2 (middle mouse button or the right mouse button on a 2-button mouse) until you select the item from the submenu displayed by the history menu item. (You do not have to hold mouse button 2 for Windows 95.) You can invoke the history pop-up from any item in the navigation list as you would the pop-up menu.
- Validate a selection item in error. The Navigation Window displays erroneous items with a yellow question mark next to the item. The validate function displays messages for any configuration errors or required fields that do not have correct values based on the list item selected. Using the validate function on a folder in the navigation list displays validation messages for all of the navigation list items within that folder.

  If you correct an error and leave the panel that contained the error, the program removes the question mark.

Changing the Configuration Program Options

Use the Options menu to change:

- Colors
- Default Model
- Default Adapter
- Message Prompting

- ATM Address Format

Notes:

1. The Multiprotocol Access Services, Multiprotocol Switched Services, and Multiprotocol Switched Services Clients versions of the configuration program do not contain the Default Adapter option.

2. The Access Integration Services version of the configuration program does not contain the ATM Address Format or Default Adapter options.

Using The Configuration Window

As you select items in the Navigation Window, the program presents panels in the Configuration Window. The format of the configuration panels depends on the item that you select.
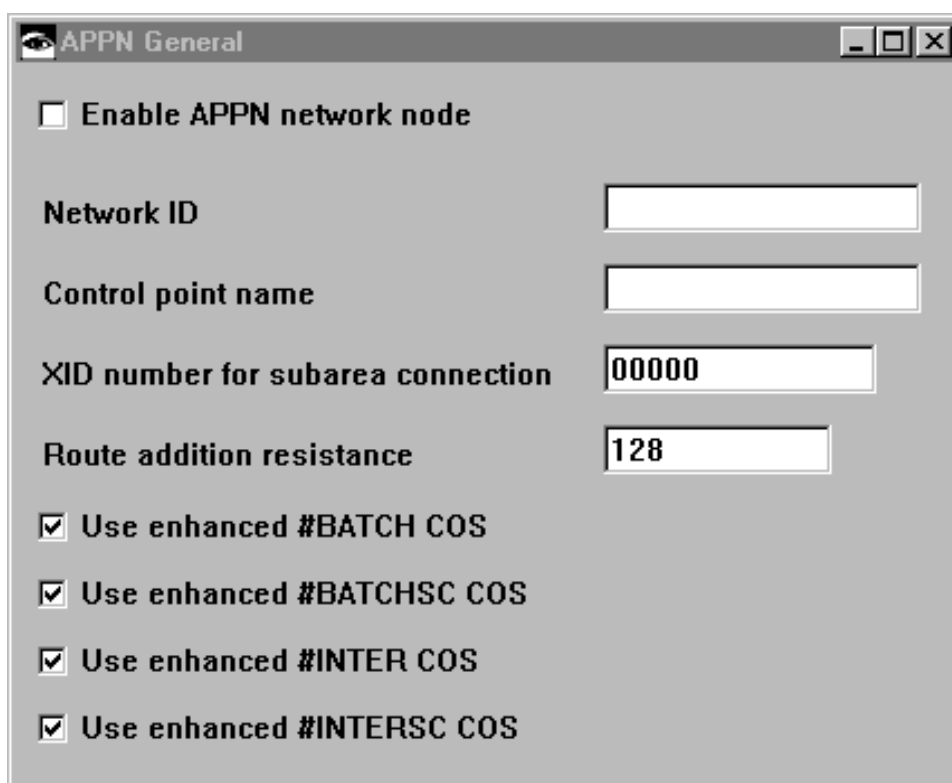
Figure 3. The Configuration Window

Navigating Through the Configuration Program

The configuration program features a point-and-click interface. You select items from lists using a pointer and then fill in items in the various panels in the Configuration Window. Various keys also have meanings specific to the configuration program.

Using the Mouse

The mouse is the primary means for moving through the various lists and panels in the configuration program.

The configuration program uses the following buttons on a 3-button mouse:

- Left (button 1) - to select an item in a panel
- Middle (button 2) - to perform the following actions:
  - On the Navigation Window
    - Display a pop-up menu for a selected item in the scrollable list.
    - Move through and select items in a menu.
    - Validate a selection item in error. The Navigation Window indicates erroneous items with a yellow question mark next to the item. The validate function displays a message for each required field that does not have a valid value and for any configuration errors.
  - On the Configuration Window
    - Drag-and-drop items in lists. The program indicates that drag-and-drop is active by changing the cursor into a hand icon. Use this function to reorder filters, dump entries, and other configuration lists. For example, you can drag and drop items in a list of IP addresses to reorder them and establish a new default gateway.
- Right (button 3) is not currently used

The left and middle mouse button descriptions are mapped to the left and right mouse buttons on a 2-button mouse.

Selecting Multiple Items in a List

If the configuration program displays a selection list, you can pick multiple items from those lists. The method for selecting items depends on whether you want to start at one point and end at another point or just choose items at random from the list.

To select multiple individual items:
Press and hold the **Ctrl** key, point to each item, and then press mouse button 1.

To select a range of items:
Select the first item in the range. Press and hold the **Shift** key. Select the last item in the range and then press mouse button 1.

To deselect items:
Press and hold the **Ctrl** key, point to the item you want to deselect and press mouse button 1. If you do not hold the **Ctrl** key, you will deselect all previously selected items except the item just selected.

Using the Configure Menu

The Configure pull-down menu enables you to access the configuration program's features. The following list describes the menu choices and their use. The underscored character and the text in parentheses indicate the keyboard shortcut for the choice.

New configuration
Reset the working configuration to the default configuration for a specific model. The program resets the Navigation Window with a list appropriate for the selected model.

Change Model

Resets the configuration program to support a different router model.

Note: This menu is not supported on the 2212 or 2216.

Open configuration (Alt+O)

Update an existing configuration or use an existing configuration as a basis for a new configuration.

Save configuration (Alt+S)

Save the current configuration into a configuration file. If the current configuration is named **config** and a configuration named config does not exist, the program invokes the "Save configuration as" dialog to allow you to specify a different name with the file extension **csf**.

Save configuration as

Save the current configuration with a different configuration name and the file extension **csf**.

Delete configuration

Remove configurations from a configuration database.

Create router configuration

Create a binary configuration file that you load into your router.

Read router configuration

Read a binary configuration file created by the "Create router configuration" option or obtained from the router using TFTP.

Communications

- Send or retrieve a configuration to or from a single router, restart a single router, or query a single router's information.
- Send or retrieve configurations to or from multiple routers or restart multiple routers.

Note: Before using the communication features, you must establish communications between the configuration program and the router. See "Establishing Communications Between the Configuration Program and a Router" for details.

ASCII file   Create an ASCII version of a configuration. You can use this file as documentation of a device's configuration.

You can also read in an existing ASCII configuration file to load into a device.

Exit (Alt+X)   Leave the configuration program.

Establishing Communications Between the Configuration Program and a Router

Before you can use the configuration program to communicate with a router, you must establish communications between the program and the router.

If you already have a configuration loaded into the router, you need perform only the steps described in "Setting Up the Configuration Program" on page 16; otherwise you must perform the steps in both "Setting Up the Router" and "Setting Up the Configuration Program" on page 16.

Setting Up the Router

If your router has no configuration loaded, you must perform the steps in this section before you can send an existing configuration to the router.

This situation would occur if:

- You have a new router with no configuration
- You cleared the router's configuration
- The router encountered a problem that cleared the configuration

To set up the router:

1. Establish access to the router. You can establish access either locally through an ASCII terminal, or remotely using Telnet, as described in your device's setup guide. To use Telnet, you must at least configure the IP address for the device (see step 2).
2. Access quick configuration. For the 2210, if the router is in EZSTART mode, enter **stop** to access QCONFIG. For other devices, if the device is in `Config Only>` mode, enter the **qc** command to access quick configuration. See the device's setup guide for more information about quick configuration.
3. Establish the IP address and IP mask for the primary interface for the router, using the following procedure.

    Note: Steps 3.a and 3.b apply only to the 2210.
    a. Reply **yes** to the Configure Interfaces question.
    b. Supply the interface information that is required for the LAN interface that will communicate with the configuration program.
    c. Reply **no** to the Configure Bridging question.
    d. Reply **yes** to the Configure Protocols question.
    e. Reply **yes** to the Configure IP question.
    f. Reply **yes** to the Configure IP on this interface question for the LAN interface that will communicate with the configuration program.
    g. Enter the IP address at the `IP Address` prompt.
    h. Enter the IP mask at the `Address Mask` prompt.
    i. Reply **no** to the Configure IP on this interface question for the other router interfaces.
    j. Reply **no** to the Enable Dynamic Routing question.
    k. Reply **yes** to the Define community with Read_Write_Trap Access question. Enter the community name that will have Read_Write_Trap access.
    l. Reply **yes** to the Save this configuration question.
4. Restart or reload the router.

When the router completes the restart sequence, it can communicate with the configuration program.

Setting Up the Configuration Program

Note: This procedure assumes that the IP address and the SNMP community have already been configured in the router. If the address and community have already been configured, you can use the retrieve function under Communications, Single, to retrieve the information from the router.

To set up the configuration program to communicate with the router:
1. Select the appropriate model from the New Configuration menu.
2. Select **Interfaces** in the IP folder.
3. In the Configuration Window:
   a. Select **IP Addresses** next to the LAN interface with which the configuration program will communicate.
   b. Enter the *IP address* and *Subnet Mask* for this interface.

      Note: These addresses must match the addresses that are configured on the router's interface.
   c. Select **Add**
4. Select **Communities** under the SNMP folder in the Navigation Window.
5. In the Configuration Window:
   a. Enter the name of the community in the Name field on the General page. Use the community defined in step 3.k on page 15.
   b. Select **Read-write trap** in the drop-down list under Access type.
   c. Select **Add**

      Note: The name and the access type must match the name and access type that is configured on the router with which the configuration program will communicate.

The configuration program can now communicate with the router and perform functions such as *Send configuration*, *Retrieve configuration*, or *Query router information.*

Note: Unless the version of the configuration program and the software loaded on the router are the same, the retrieve function may fail. Use the procedure described in "Upgrading/Migrating Previous-Level Configuration Files" on page 7 to migrate the configuration from the previous version.

Using the Command Line Facility

The Command Line facility enables you to automate configuration program operations. You can use the **config** command to enter a complete set of commands into an argument file named cfgargs, which the configuration program uses to direct its operation, or you can use it to enter commands on the command line. (See Figure 4 on page 17 for an example of a cfgargs file.) To use only the command line facility, without enabling the GUI interface, the last command in the file or on the command line must be **-exit**. If you do not meet this requirement, the program displays the GUI interface's Navigation and Configuration windows.

Note: You can use the **config** command only in a windowed operating system.

CFGARGS File Example

If you enter the **config** command, the program looks for a file named cfgargs.
Figure 4 shows the contents of a cfgargs file that:

- Opens a modified ASCII configuration file.
- Sends the open configuration to a router.
- Causes the current configuration to become the active configuration by rebooting
  the router immediately.
- Retrieves the current configuration from a router.
- Saves the current configuration into a configuration file within a configuration
  database.

```
-displayMessages off 1
-storeMessages on cfgMessages.log 2
-asciiRead routerOne.acf withoutLogging 3
   —send 1.1.1.1 private 4
   —reboot 1.1.1.1 private 5
-retrieve 2.2.2.2 private 6
   —save * routerTwo 7
-exit
```

*Figure 4. Example of CFGARGS File Contents. An entire set of commands may be entered
as one string, but using a separate line for each command makes the file easier to read and
maintain.*

1. Turns off message display. If any message offers a choice, takes the default
   choice.
2. Logs all messages to the specified log file.
3. Opens the specified ASCII file.
4. Sends the configuration to a router.
5. Causes the configuration to become the active configuration by rebooting the
   router immediately.
6. Retrieves the configuration from the specified router.
7. Stores the configuration into the database that was specified by the previous
   save command, using the specified configuration name.

Specifying the Config Command

You may enter the **config** command with or without additional commands and their
arguments:

config   Causes the program to search for a cfgargs file in the configuration
program directory and, if the file exists, performs the commands in
it. When the program finishes processing the commands, it ends
and deletes the file. If a cfgargs file does not exist, the program
waits for additional commands on the command line.

> **Note:** To facilitate making future configuration changes, you may
> want to prevent automatic deletion of the cfgargs file. You
> can do this by using a text editor to open the cfgargs file and
> keeping it open while the configuration program processes
> the commands in it. As an alternative, you may want to save
> a copy of the file under a different name.

config -*command* **[***args***] [...]**

If you specify additional commands and arguments on the command line, the program processes them. Each command must start with a minus sign (–) and be followed by as many additional arguments as needed.

If an argument, *args*, contains embedded blanks, you must enclose the argument in double quotes (" ").

Note: If you plan to enter commands on the command line, you must ensure that a cfgargs file does not already exist in the configuration program directory. This is necessary because if one does exist, the program processes only the commands in it and ignores the commands you enter on the command line.

The following list describes the **config** command-related commands and arguments.

Notes:

1. Commands and arguments must be separated from each other by either spaces, tabs, or new lines.
2. The MSS and MSSC versions of the configuration program do not support the **-asciiRead** or **-asciiWrite** commands.

-asciiRead *filename* [**withoutLogging**]

filename Required argument. Specifies the name of an ASCII configuration file and reads it into the configuration program as the current configuration in memory. Note: For AIX, if the *filename* contains embedded blanks, you must enclose it in double quotes (" ").

withoutLogging
Optional argument. Specifies that validation errors should not be logged.

Note: By default, the configuration program writes validation errors to an ASCII file, which has the same name as the file you read, but with a **.vlg** extension.

-asciiWrite *filename* [**withComments**]

filename Required argument. Specifies the name of the ASCII configuration file and creates the file from the configuration program's current configuration in memory.

withComments
Optional argument. Indicates whether the output file should contain extra commentary describing the ASCII configuration file.

-commandsFrom
> *filename*
>
> | *filename* | Required argument. Specifies the name of a file containing additional commands, and reads and processes those commands. |
> |---|---|

-createConfiguration
> *filename*
>
> | *filename* | Required argument. Specifies the name of the file being created (in the binary format recognized by the router) from the program's current configuration in memory. |
> |---|---|

-displayMessages
> **on** | **off**
>
> | **on** | Enables the displaying of messages. Allows you to control the displaying of messages. |
> |---|---|
> | **off** | Disables the displaying of messages. If a message would have offered you a choice of actions, the program takes the default action. |

Notes:

1. The **-displayMessages** command can appear anywhere in the file.
2. You can have multiple **-displayMessages** commands in a file, but the configuration program only honors the last one.

-exit	Closes the configuration program's GUI windows and ends the program. If you specify multiple commands, **-exit** must always be the last one.

-exitOnError	If an error occurs while processing the command line, ends the program without starting the GUI. The errors.log file identifies the source of the error.

-noLogo	Suppresses displaying the configuration program's logo window during startup. Because the logo window requires you to select the OK button, specify this command to run the configuration program in non interactive mode.

-open	*directory config*

> | *directory* | Required argument. Specifies the name of the directory containing the configuration file you want to open. |
> |---|---|

| | *config* | Required argument. Specifies the name of the configuration file you want to open. The command stores the contents as the program's current configuration in memory. If the file name contains embedded blanks, you must enclose the name in double quotes (" "). |
| | | |

-queryinfo *host name* | *ip address community* [*timeout*]

           host name **|** *ip address*

	host name **|** *ip address*	Required argument. Specifies either the name of the router, which the program can resolve into an IP address, or the router's actual IP address. The command retrieves a configuration summary from the router. The program displays and logs the summary based on the setting of the **-displayMessages** and **-storeMessages** commands, respectively.
	community	Required argument. Specifies an SNMP community that is defined on the router with read access.
	timeout	Optional argument. Specifies the time (in seconds) before the SNMP request expires.

-read *filename*

	filename	Required argument. Specifies the name of the router configuration file to be read, and reads it into the program's current configuration. The file must be in the binary format recognized by the router.

-reboot *host name* | *ip address community* [*timeout*] [*date*] [*time*]

	host name	*ip address*	Required argument. Specifies either the name of the router, which the program can resolve into an IP address, or the router's actual IP address. The command sends a restart request to the router at the specified date and time, based on the date and time of the workstation on which the program is running. The program calculates the number of seconds between the date and time the program processes the command and the date and time specified in the command arguments.
	community	Required argument. Specifies an SNMP community that is defined on the router with read access.

	timeout	Optional argument. Specifies the length of time (in seconds) before the SNMP request expires.
	date	Optional argument. Specifies the date on which the router will reboot. The default is the current date.
	time	Optional argument. Specifies the time at which the router will reboot. The default is the current time.
-retrieve	*host name	ip address community* [*timeout*]	
	*host name	ip address*	Required argument. Specifies either the name of the router, which the program can resolve into an IP address, or the router's actual IP address. The command retrieves the configuration from the router into the program's current configuration in memory.
	community	Optional argument. Specifies an SNMP community that is defined on the router with read access.
	timeout	Optional argument. Specifies the length of time (in seconds) before the SNMP request expires.
-save	*directory config*	
	directory	Required argument. Specifies the name of the directory containing the configuration file into which you want to save the current configuration in the program's memory. If you specify an * instead of a name, the program uses the most recently named configuration storage directory.
	config	Required argument. Specifies the name of the configuration file into which you want to save the current configuration. The default is the current configuration name.
-send	*host name	ip address community* [*timeout*]	
	*host name	ip address*	Specifies either the name of the router, which the program can resolve into an IP address, or the router's actual IP address. The command sends the program's current configuration in memory to the router.
	community	Optional argument. Specifies an SNMP community that is defined on the router with read access.
	timeout	Optional argument. Specifies the length of time (in seconds) before the SNMP request expires.

-sendMultiple *send list*

> *send list*   Required argument. The send list contains the list of configurations and router addresses. You generate this list using the configuration program menu bar to select **Options** → **Communications** → **Send Multiple**, and completing the fields in the Send Multiple window. This command sends multiple router configurations to multiple routers.

-storeMessages
> **on** [*filename*] **off**

> **on**   Enables the logging of messages. If a log file exists, the system appends messages to it. If a log file does not exist, the program creates one.

> *filename*   Optional argument. Specifies the name of the file in which to put logged messages. The default is **cfg.log**.

> **off**   Disables the logging of messages.

Notes:

1. The **-storeMessages** command can appear anywhere in the file.
2. You can have multiple **-storeMessages** commands in a file, but the configuration program only honors the last one.
3. If you disable **-displayMessages** and enable **-storeMessages**, the program writes all messages to the log file.

-timedcfg   *host name | ip address community* [*timeout*] [*date*] [*time*]

> *host name | ip address*
> Required argument. Specifies either the name of the router, which the program can resolve into an IP address, or the router's actual IP address. The command sends a configuration file to the router with a request for reload at the specified date and time, based on the date and time of the workstation on which the program is running.

> **Notes:**

> 1. This command is similar to the **-send** command except that a timer is set, based on the values provided in the date and time arguments. The timer is not affected by reloads or power outages. When the timer value expires, the router loads the configuration.
> 2. If the date and time arguments specify something earlier than the current date and time, the configuration is sent to the router's

| IBD (FLASH), but the timer is not set and the
| configuration will never be loaded.

| *community*   Required argument. Specifies an SNMP community
| that is defined on the router with read access.

| *timeout*   Optional argument. Specifies the length of time (in
| seconds) before the SNMP request expires.

| *date*   Optional argument. Specifies the date on which the
| router will reload. The default is the current date.

| *time*   Optional argument. Specifies the time at which the
| router will reload. The default is the current time.

Creating and Using ASCII Files

By default, the configuration program saves configuration files in binary format. You
can use the configuration program to create and read configuration files in ASCII
format. You can also convert configuration files from one format to the other.

An ASCII configuration file may be useful if you need to alter many configurations at
one time without having to load configurations into the graphical user interface.

Note: The ASCII configuration facility is not intended to be used to create new
configurations or to make major modifications to existing configurations.

For example, if you need to update many IP addresses throughout a network, it
may be quicker to save the affected configuration files in ASCII format and then
manually (or automatically with a program) update the addresses. A configuration
file saved in ASCII format is also useful when you want to scan the configuration
online or print the configuration for review.

After updating an ASCII configuration, you can read it into the configuration program
and save the configuration in binary. You can load a configuration into a
multiprotocol device only in the binary format.

Understanding the ASCII Functions

The ASCII functions are contained in the Configure menu pull-down under the
ASCII file item. When you select **ASCII file** you see the following drop-down menu
items:

- **Create single file...** creates a single ASCII configuration file.
- **Read single file...** reads a single ASCII configuration file into the configuration
  program's memory.

If you select **Create single file...**, the program displays a dialog in which you
specify the path and file name to which the program writes an ASCII file. You can
also print comments in the ASCII File from this dialog, if desired.

If you select **Read single file...**, the program displays a dialog in which you specify
the path and file name from which the program reads an ASCII file. You can also
prevent the configuration program from writing validation errors to a validation file in
this dialog. By default, the validation file has a **.vlg** file extension.

Note: For the **Read single file...** function to read an ASCII file successfully, you must either:
- Create the file with the **Create single file...** function, or
- Create an ASCII file with the same parameter structure that the **Create single file...** creates.

"Processing ASCII Files Using the Command Line Facility" on page 30 explains how to use the configuration command line facility to process ASCII files.

Creating an ASCII File

The configuration program uses the current configuration to create an ASCII file. Therefore, if you want to create an ASCII file for a particular device:

1. Configure the device. If you require an ASCII file for a device that is already configured, use the **Retrieve configuration** function to obtain the configuration and load it into the program's memory.
2. Select **Configure** from the Navigation Window menu.
3. Select **ASCII file** from the drop-down menu but do not release the mouse button at this time.
4. Move to the right and select **Create single file...**. The program displays a file dialog.
5. Enter the path and file name to which the program is to write the ASCII file. If you want to replace an ASCII file, you can select an existing file from the file dialog.
6. Select **OK**.

The configuration program uses the current configuration in memory to CREATE the ASCII file. This file will contain the configuration parameters and their current values.

Creating an ASCII file is effectively the same as converting a configuration from its binary form to ASCII. "Interpreting an ASCII File" on page 26 explains the contents of a sample of an ASCII file.

Adding Comments to an ASCII File

You can add comments to an ASCII file. These comments will help you to use the ASCII file as a template for other configurations. The comments include:
- A brief description of the parameter
- The type of value
- The valid values or value ranges, if applicable

The comments are a guide to help you update the ASCII configuration file. "Using an ASCII File as a Configuration Template" on page 32 explains using ASCII files as configuration templates.

Reading an ASCII File

> **Important**
>
> Because configuration parameters change between releases, the configuration program will read only ASCII files created by the same version configuration program. Altering the version number of an older ASCII file or changing the version number in a valid configuration file will cause the read function to fail. See "Understanding How the Configuration Program Validates ASCII Files" on page 32 for an explanation of how the configuration program validates an ASCII configuration.
>
> You cannot upgrade an ASCII file from one version of the configuration program to another version. You can only migrate binary configuration files (**.csf**) between releases. For the procedure to migrate older configuration files, see "Upgrading/Migrating Previous-Level Configuration Files" on page 7.

When you read an ASCII file, the configuration program reads the file into memory and replaces the current configuration. If you need the configuration currently in memory, use the **Save configuration** or **Save configuration as** functions before reading in the ASCII file. See "Understanding the ASCII Functions" on page 23 for read function restrictions.

See "Interpreting an ASCII File" on page 26 for a description of the ASCII file that can help you create your own ASCII file.

To read an ASCII file:

1. Select **Configure** from the Navigation Window menu.
2. Select **ASCII file** from the drop-down menu but do not release the mouse button at this time.
3. Move to the right and select **Read single file...**. The program displays a file dialog.
4. Enter the path and file name or select a file in the dialog from which the program should read the ASCII file.
5. If you want the configuration program to create an ASCII record of errors in the configuration, select **Log validation errors**, which is the default. Because the configuration program always validates configurations and shows errors, you do not need to log validation error, but the feature is useful:
   * To check the configuration and correct any problems before transmitting the binary file to the device.
   * When reading ASCII files from multiple routers so that you can determine any validation errors when the read process completes.

   Note: If you make changes in the configuration program, you can use **Create single file...** to replace the erroneous ASCII file.
6. Select **OK**.

By default, the validation file has a file extension of **.vlg**.

After reading in the configuration, you can:
* Send the configuration file to a device using **Send configuration to router**.
* Save the configuration using **Save configuration** or **Save configuration as**.

Converting from ASCII to Binary

Multiprotocol devices require that the configurations be in binary. Using the configuration program, you can convert an ASCII file to binary by:

1. Reading in the ASCII file as described in "Reading an ASCII File" on page 25 and sending the configuration to the device using the **Send** function.

2. Reading in the ASCII file as described in "Reading an ASCII File" on page 25 and creating a new configuration file using the **Create configuration** function.

3. Reading in the ASCII file as described in "Reading an ASCII File" on page 25 and saving the configuration using the **Save configuration** or **Save configuration as** functions.

Converting from Binary to ASCII

If you have a binary configuration file, you can convert it by:

1. Reading the file, which by default has a **.csf** extension, into the configuration file using **Open configuration** and creating the ASCII file using **Create single file...**.

2. Retrieving a configuration from a device and creating the ASCII file using **Create single file...**.

Interpreting an ASCII File

Figure 5 on page 27 contains a sample of a portion of an ASCII file with comments. When interpreting the ASCII file, note that:

- Header information begins with an exclamation mark (!).
- Comments begin with an asterisk (*) in column 1. The comments contain descriptions of the parameters and the valid values and ranges. When necessary, the comments indicate the type of the parameters. The types are:

Multiple Segments	Used for data composed of well-defined parts. For example, a four-segment numeric address that is used to define IP addresses and IP address masks.
Numeric	Typically used for data that may have mathematical calculations performed on it. Data value ranges are enforced on all numeric data types.
String	Each string must be within a certain length and can be composed only of a specific set of characters.
Enumerated	Used for multiple-choice values. When an ASCII file is saved with comments, the comments explain the possible values. For example:

        Enable/Disable
                1 = enable, 2 = disable. Used mainly for enabling/disabling adapter ports and protocols.

- Configuration groups and parameters begin with a pound sign (#).

- A group of items is contained within { and }.
- Elements in a list of items are contained within { and }.
- The word ″nil″ signifies that the parameter currently has no value.
- All parameters appear in the file even if you did not configure them. When you did not explicitly specify the value, the program uses the default value. If a default value for the parameter does not exist, the program uses "nil".
- Encrypted values are preceded by an "E." To specify a new value to be encrypted by the configuration program, change the "E" to a "U" and overwrite the old value with the new value. When the configuration program reads in the ASCII file, it will encrypt the new value.

```
2210 Configuration
November 18, 1996 at 3:08:56 pm
!  'Configuration:' = 'config'
!  'Model:' = 'model 121/T'
!  'Adapter:' = 'emptySlot'
!  'Version:' = '13011'

#'appletalk' = {

*   at2General = {
*      at2Enable : Enable Appletalk 2
*        value : enable or disable
*      at2Checksum : Checksum
*        value : enable or disable
*      at2Translation : Translate
*        value : enable or disable
*      at2FastPath : Fast path cash size
*        value : range from 100 to 10000   }
    #'at2General' = {
    #'at2Enable' = 'disable'
    #'at2Checksum' = 'disable'
    #'at2Translation' = 'disable'
    #'at2FastPath' = '500'
  }

*   at2Interfaces = { list of at2Interface }
*   at2Interface = {
*      at2InterfaceGeneral = {
*         at2FirstNetworkRange : Lowest number of network range
*           value : range from 1 to 65279
*         at2LastNetworkRange : Highest number of network range
*           value : range from 1 to 65279
```

Figure 5. Sample ASCII Configuration with Comments (Part 1 of 4)

```
*           at2NodeNumber : Node number
*             value : range from 0 to 254
*           at2DefaultZoneName : Default zone name
*             value : a string no more than 34 character(s) in length
*                    with any ASCII character
*           at2InNFilter : Input NFilter
*            value : 1 of the following strings: None Inclusive Exclusive
*           at2OutNFilter : Output NFilter
*           value : 1 of the following strings: None Inclusive Exclusive
*           at2InZFilter : Input ZFilter
*             value : 1 of the following strings: None Inclusive Exclusive
*           at2OutZFilter : Output ZFilter
*             value : 1 of the following strings: None Inclusive Exclusive
*           at2EnableInterface : Enable interface
*            value : enable or disable
*            at2InputNetworkFilter : Input NFilter
*             value : 1 of the following strings: None Inclusive Exclusive
*            at2OutputNetworkFilter : Output NFilter
*             value : 1 of the following strings: None Inclusive Exclusive
*           at2InputZoneFilter : Input ZFilter
*             value : 1 of the following strings: None Inclusive Exclusive
*           at2OutputZoneFilter : Output ZFilter
*             value : 1 of the following strings: None Inclusive Exclusive
*           at2SplitHorizon : Split horizon
*             value : enable or disable    }
*        at2ZoneNames = { list of at2ZoneName }
*        at2ZoneName = {
*           at2ZName : Zone name
*             value : a string
*                    no less than 1 character(s) in length
*                    and no more than 32 character(s) in length
*                    with any ASCII character  }
*        at2NFilters = { list of at2NFilter }
*        at2NFilter = {
*           at2FirstFilterNetworkRange : First network range number
*             value : range from 1 to 65279
*           at2LastFilterNetworkRange : Last network range number
*             value : range from 1 to 65279
*           at2NetworkFilterType : Filter type
*             value : 1 of the following strings: In Out  }
*        at2ZFilters = { list of at2ZFilter }
*        at2ZFilter = {
*           at2FilterZoneName : Zone name
*             value : a string
*                    no less than 1 character(s) in length
*                    and no more than 30 character(s) in length
*                    with any ASCII character
```

Figure 5. Sample ASCII Configuration with Comments (Part 2 of 4)

```
*          at2ZoneFilterType : Filter type
*            value : 1 of the following strings: In Out  }    }
    #'at2Interfaces' = {
      #'at2Interface' = {
        #'at2InterfaceGeneral' = {
          #'at2FirstNetworkRange' = nil
          #'at2LastNetworkRange' = nil
          #'at2NodeNumber' = '0'
          #'at2DefaultZoneName' = nil
          #'at2InNFilter' = 'None'
          #'at2OutNFilter' = 'None'
          #'at2InZFilter' = 'None'
          #'at2OutZFilter' = 'None'
          #'at2EnableInterface' = 'disable'
          #'at2InputNetworkFilter' = 'None'
          #'at2OutputNetworkFilter' = 'None'
          #'at2InputZoneFilter' = 'None'
          #'at2OutputZoneFilter' = 'None'
          #'at2SplitHorizon' = 'enable'
        }

        #'at2ZoneNames' = {   }
        #'at2NFilters' = {   }
        #'at2ZFilters' = {   }
      }
      #'at2Interface' = {
        #'at2InterfaceGeneral' = {
          #'at2FirstNetworkRange' = nil
          #'at2LastNetworkRange' = nil
          #'at2NodeNumber' = '0'
          #'at2DefaultZoneName' = nil
          #'at2InNFilter' = 'None'
          #'at2OutNFilter' = 'None'
          #'at2InZFilter' = 'None'
          #'at2OutZFilter' = 'None'
          #'at2EnableInterface' = 'disable'
          #'at2InputNetworkFilter' = 'None'
          #'at2OutputNetworkFilter' = 'None'
          #'at2InputZoneFilter' = 'None'
          #'at2OutputZoneFilter' = 'None'
          #'at2SplitHorizon' = 'enable'
        }
        #'at2ZoneNames' = {   }
        #'at2NFilters' = {   }
        #'at2ZFilters' = {   }
      #'at2Interface' = {
        #'at2InterfaceGeneral' = {
          #'at2FirstNetworkRange' = nil
          #'at2LastNetworkRange' = nil
          #'at2NodeNumber' = '0'
          #'at2DefaultZoneName' = nil
          #'at2InNFilter' = 'None'
          #'at2OutNFilter' = 'None'
```

Figure 5. Sample ASCII Configuration with Comments (Part 3 of 4)

```
                         #'at2InZFilter' = 'None'
                         #'at2OutZFilter' = 'None'
                         #'at2EnableInterface' = 'disable'
                         #'at2InputNetworkFilter' = 'None'
                         #'at2OutputNetworkFilter' = 'None'
                         #'at2InputZoneFilter' = 'None'
                         #'at2OutputZoneFilter' = 'None'
                         #'at2SplitHorizon' = 'enable'
                       }
                     #'at2ZoneNames' = {  }
                     #'at2NFilters' = {  }
                     #'at2ZFilters' = {  }
                   }
                }
             }
     .
     .
     .
```

Figure 5. Sample ASCII Configuration with Comments (Part 4 of 4)

Processing ASCII Files Using the Command Line Facility

Besides the functions described in "Understanding the ASCII Functions" on page 23, the configuration program command line facility also supports the following functions:

Convert Binary to ASCII
Enables you to select one or more binary formatted configuration files from a configuration database, create ASCII formatted files, and save them to the hard drive.

Convert ASCII to Binary
Enables you to select one or more ASCII formatted configuration files from the hard drive, create binary formatted configuration files, and save them to a configuration database.

The following sections describe using the configuration program command line facility to process ASCII files. Examples are included to help you build **cfgargs** files so you can run certain processes in batch mode.

Creating an ASCII File

Use the **asciiWrite** command to create an ASCII file. If you want to create an ASCII file for a specific device, retrieve the configuration from the device before the write command in the **cfgargs** file. The following example creates an ASCII file of the configuration for device CON656A with comments and then exits the configuration program:

```
-displayMessages off
-storeMessages on cfgCommands.log
  -retrieve CON656A myread
  -asciiWrite d:\asccfgs\con656a.acf with Comments
-exit
```

Reading an ASCII File

Use the **asciiRead** command to read an ASCII file into the configuration program's memory. The following example reads the ASCII file named rtr656b.acf in directory asccfgs:

```
-displayMessages off
-storeMessages on cfgCommands.log
  -asciiRead d:\asccfgs\rtr656b.acf
-exit
```

Note: Validation errors are logged to file rtr656b.vlg.

Converting Between ASCII and Binary

Use the **createConfiguration** command to convert an ASCII file to a binary file that you can use as backup for a device's configuration. The following example reads in an ASCII file, transmits it to a device, and then creates a binary configuration file for backup.

```
-displayMessages off
-storeMessages on cfgCommands.log
  -asciiRead d:\asccfgs\mss656c.acf
  -send mss656c mywrite
  -createConfiguration d:\bincfgs\mss656c.bin
-exit
```

Converting a Retrieved File to ASCII

There may be times when you need to review the current configuration, for example, debugging or planning for an upgrade. The next example retrieves the configuration from device RTR656D and converts the configuration to ASCII for later use.

```
-displayMessages off
-storeMessages on cfgCommands.log
  -retrieve rtr656d readit
  -asciiWrite d:\asccfgs\rtr656d.acf
-exit
```

Using the ASCII File Functions with Multiple Devices

There may be times when you need to either create ASCII versions of configurations from multiple routers or convert multiple ASCII files to binary to transmit to various devices. The following sections outline a method to achieve those goals using the command line facility.

Creating Multiple ASCII Files: This following example assumes that you have just received five devices that were pre-configured. It illustrates a way to create documentation of the configuration for those devices.

```
-displayMessages off
-storeMessages on cfgCommands.log
  -retrieve rtr656a reada
  -asciiWrite d:\asccfgs\rtr656a.acf
  -retrieve rtr656b readb
  -asciiWrite d:\asccfgs\rtr656b.acf
  -retrieve rtr656c readc
  -asciiWrite d:\asccfgs\rtr656c.acf
  -retrieve rtr656d readit
  -asciiWrite d:\asccfgs\rtr656d.acf
  -retrieve rtr656e readit
  -asciiWrite d:\asccfgs\rtr656e.acf
-exit
```

Reading Multiple ASCII Files: The following example assumes that you need to reload five devices with configurations and you have valid ASCII files. You have

already set up the devices to communicate with the configuration program on each of the devices. Each device has a unique configuration and you do not need the binary configuration saved.

```
-displayMessages off
-storeMessages on cfgCommands.log
  -asciiRead d:\asccfgs\rtr656f.acf
  -send mss656f mywritef
  -asciiRead d:\asccfgs\rtr656g.acf
  -send mss656g mywriteg
  -asciiRead d:\asccfgs\rtr656h.acf
  -send mss656h mywriteh
  -asciiRead d:\asccfgs\rtr656i.acf
  -send mss656i mywritei
  -asciiRead d:\asccfgs\rtr656j.acf
  -send mss656j mywritej
-exit
```

Using an ASCII File as a Configuration Template

The ability to create and read ASCII files can assist you in automating the process of creating or updating a large number of device configurations. The following procedure outlines a method that you can use to perform a mass update.

1. Create a file that contains the new information for the devices you are configuring. You can use a full or partial template. Make sure that any interfaces or adapters specified match the models and adapters specified in the header.

2. Create a program to read the device file and a configuration template file.

3. Create a copy of the template file, update the new copy with the device information, and save the file with a name that would help you to identify the device for later transmittal.

4. Locate a device in your network that contains a configuration that closely matches the target devices. Ideally, the only things you should need to change are items like IP addresses, MAC addresses, dial circuit numbers, or other such items. The devices should support and use the same filters and protocols and have the same interfaces.

5. From the device you located in step 4, retrieve the device configuration using **Retrieve configuration**. You can retrieve the configuration with the command line facility of the configuration program.

6. Create an ASCII file of the configuration.

7. Create the new configurations using the ASCII file you created in step 6, the file you created in step 1, and the program you created in step 2.

After you create the new configurations, you can use the configuration program to read the resulting ASCII files, save them as binary files, if needed, and then send the files to the appropriate devices. "Reading Multiple ASCII Files" on page 31 contains more information about how the command line facility can help you automate this task.

Understanding How the Configuration Program Validates ASCII Files

The configuration program performs the following actions while reading the ASCII files:

- If there is an adapter specified in the header, the program compares that value with the adapter type in the Devices group of the configuration file.
- If there is a model specified in the header, the program:

1. Checks to see that the defined interfaces match the model capabilities. For example, a 2210 Model 121 or 12T cannot have an Ethernet interface. In this case, the program discontinues reading the configuration and logs an error.

2. Verifies that the number of interfaces is correct for the device configuration identified in the header.

    If there are too many interfaces defined, the program discontinues reading the configuration because it cannot determine which definitions are correct and logs an error.

    If there are too few interfaces, the program fills in defaults for the undefined interfaces. For example, if the device has two WAN ports and you supply values for one, the program uses defaults for the missing interface after the configured interface.

    Note: Undefined interfaces must be the last interfaces in the interface list. For example, if you define interfaces 0, 2, and 3, the configuration program will not use defaults for the missing interface. In this case, you must define the consecutive interfaces 0, 1, 2, and 3 for the program to provide defaults for the missing interface. For interface 1, you *must* provide a default interface.

* Compares the number of device interfaces to the number of elements in the interface protocol list.

    If there are fewer elements in the interface protocol list than the number of device interfaces, the program fills the list with default protocols for the remaining interfaces.

    If there are more elements in the interface protocol list, than the number of device interfaces, the program discontinues reading the configuration and logs an error.

Configuring a Router Using the Configuration Program

When you start the configuration program, it displays three windows:

   The Copyright Notice window
   The Navigation window
   The Configuration window

Select **OK** on the copyright notice window to begin using the configuration program.

Configuring a New Router

To configure a router using the configuration program:

1. Select **Configure** from the Navigation Window.
2. Select the model you wish to configure from the **New configuration** submenu.
3. Select **Adapter Interfaces** from the Devices folder and configure any adapters for the router in their appropriate slots.

    Notes:
    a. There is a Ports panel in the IBM Nways Access Integration Services, Multiprotocol Access Services, Multiprotocol Routing Services, Multiprotocol Switched Services, and Multiprotocol Switched Services Clients and IBM Nways Multiprotocol Access Services versions of the configuration program. This panel does not exist for the IBM Nways Multiprotocol Routing Services or the IBM Nways Multiprotocol Switched Services versions.

b. The IBM Nways Multiprotocol Routing Services version of the configuration program displays the physical ports first followed by the virtual ports.

In the IBM Nways Access Integration Services, Multiprotocol Access Services, Multiprotocol Routing Services, Multiprotocol Switched Services, and Multiprotocol Switched Services Clients and IBM Nways Multiprotocol Access Services versions, the virtual ports can be intermingled with the physical ports.

c. The Interfaces panel displays the interfaces sorted by slot and then port.

4. Select **Interfaces** from the Devices folder and configure the interfaces for the router.

5. Expand the General folder under the Devices folder, if not already expanded.

6. Select the items in the General folder and configure the parameters in each of those items.

7. Select the items in the System folder and configure the parameters in each of those items.

8. Select and configure the remaining folders as necessary.

Changing an Existing Configuration

Note: When changing an existing configuration using the configuration program, make sure that other personnel do not update the configuration using the command line interface. Any configuration data entered through the command line is lost when you send the configuration to the router from the configuration program.

To change an existing configuration:

1. Retrieve the configuration from the router you are changing.

2. Use the configuration program to change the parameters you need to change.

3. Save the configuration.

4. Send the configuration to the router using the **Communications** menu item.

Using Guided Configuration

The configuration program contains guided configuration to help you configure the device's features and protocols. The guided configuration consists of elementary procedures that step you through the configuration process. To access guided configuration:

1. Select a folder in the Navigation Window.

2. Press **F1** to display the text of the procedure in a help window.

You can now follow the process to configure the feature or protocol.

Notes:

1. There may be multiple ways to configure a feature or protocol. The guided configuration is a recommended procedure.

2. The guided configuration does not remember where you were in the procedure. If you follow a link to other help information from the procedure, the help window displays the beginning of the procedure when you return to the procedure.

Correcting Configuration Program Problems

If you experience problems with the configuration program, locate the symptom in Table 2 that most closely resembles the problem and perform the actions described in the Recommended Actions column.

Table 2. Common Symptoms of Configuration Program Problems

Symptom	Recommended Actions
Cannot read or make selections from a panel The configuration program is hung	Revert to the original initialization files. Locate the file named in the following list on the CD-ROM and copy it over the existing config.ini file in the configuration program's directory. **Operating System** **Initialization File** **AIX** aix.ini **Windows** WIN.INI

Internal Execution Error or Run Time Error	Take the following actions:
	
	1. Copy the following files in the configuration program's directory to a safe place:
	   • walkback.log
	   • The binary router configuration file (**.cfg**) you were making
	   • The configuration file (**.csf**) that contains the configuration you were making
	2. Contact Customer Service.
	3. Provide the saved files to Customer Service as problem documentation.

Removing the Configuration Program

Before removing the configuration program, you must save all of the configuration files with a **.csf** extension into the configuration program's directory, if the configurations are still required for your network. Use either the **Save configuration** or **Save configuration as** choices on the Configuration menu (described in "Using the Configure Menu" on page 13).

To remove the configuration program:

1. Copy the configurations to another directory, if necessary.
2. Delete all the files in the configuration program's directory using:
   • The **rm** command on AIX
   • The **del** or **erase** command on Windows or OS/2
3. Delete the folder in Windows or OS/2.

Index

37

Readers' Comments — We'd Like to Hear from You

Nways AIS/MAS/MRS/MSS/MSSC
Configuration Program User's Guide for Nways Multiprotocol and Access Services Products

Publication No. GC30-3830-05

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you?   ☐ Yes   ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.

IBM®

Fold and Tape **Please do not staple** Fold and Tape
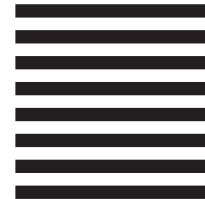
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Design & Information Development
Department CGF/Bldg. 656
PO Box 12195
Research Triangle Park, NC 27709-9990

Fold and Tape **Please do not staple** Fold and Tape

IBM®

Part Number: 02L2394

GC30-3830-05

02L2394