

# How NetEx/IP™ Works with Netfinity®



6420 Sycamore Lane North #300  
Maple Grove, MN 55369  
[www.netex.com](http://www.netex.com)

# How NetEx/IP™ Works with Netfinity®

The Netfinity, which has recently been rebranded by IBM as “IBM **eserver** xSeries,” is an IBM Intel-based platform that is used as an enabling device for NetEx/IP implementations. It is a PCI-based platform that comes in a variety of processor sizes and configurations. The model used in the NetEx/IP implementations is the Netfinity 5100 (IBM **eserver** xSeries 230).

There are four variations of NetEx/IP software for the Netfinity. Collectively, this software is referred to as NESiGate. Each platform has one of the following NESiGate products installed:

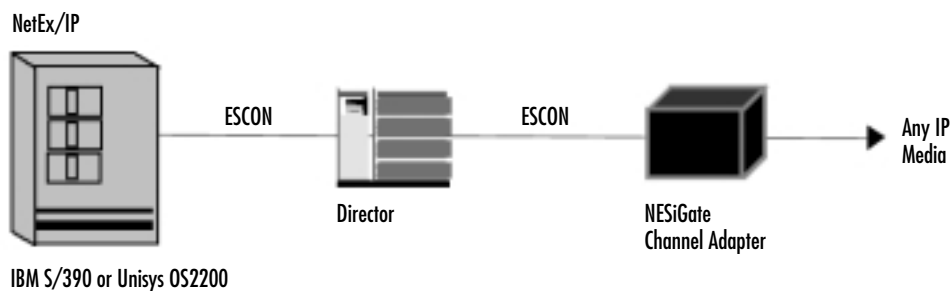
- NESiGate Channel Adapter
- NESiGate Channel Offload Adapter
- NESiGate LAN Offload Adapter
- NESiGate HYPERchannel Gateway Adapter

RedHat Linux 6.2 is the operating system used for each of the NESiGate products.

For all NESiGate implementations, changes to NetEx/IP have been made in the lower layers, which has allowed all of the NetEx Application Programming Interfaces (API's) to remain intact. This means there are no requirements to change applications in order to make them run over NetEx/IP networks.

## NESiGate Channel Adapter

This implementation is used with the channel-attached NetEx/IP host products for IBM OS/390 (H210IP) and Unisys (H300IP). By using the Netfinity as a channel-to-IP network gateway, H210IP and H300IP are able to communicate with other NetEx/IP hosts over standard IP-based infrastructures, such as Fast Ethernet, Gig Ethernet or FDDI, without requiring network connectivity on the IBM or Unisys hosts. The only host connectivity requirement is to have either an ESCON or Parallel channel to provide the connectivity to the Netfinity.



Since the Netfinity provides all the network connectivity, an additional benefit of this solution is that it allows network connections to be accessed from systems that may not natively provide support for that particular network interface. For example, the Netfinity could provide a channel gateway interface to a Gig Ethernet backbone on systems that otherwise may not be able to physically connect to that backbone.

## **How It Works**

A NESiGate Channel Adapter consists of up to two channel interfaces (ESCON and/or Parallel), as well as one or more standard network interfaces. The NESiGate software, working in conjunction with the host based NetEx/IP, interfaces between the channel and the IP router, and provides the device-to-network address mapping.

The NESiGate Channel Adapter software consists of the following components:

- Channel Interface
- IP Interface
- Router
- Control Program
- DNS Requester
- Operator Interface
- Web Browser Interface
- ESCON Channel (optional)
- Parallel Channel (optional)

## **Channel Interface**

The channel component executes in the Linux kernel and provides the interface between the Router component and the channel firmware residing on the ESCON or Parallel channel board.

At least two devices are required by NetEx/IP — one for input and one for output. More devices can be used if connections are established to multiple remote NetEx/IP hosts. The devices are defined as a 3174 device type on IBM OS/390 systems, and as an Arbitrary Device on Unisys OS2200 systems.

## **IP Interface**

The IP interface component executes in the Linux kernel and provides the interface between the Router component and the Linux TCP/IP stack. It interfaces with TCP/IP by using UDP socket calls to deliver data to the network.

## Router

The Router component executes in the Linux kernel and provides the interface between the Channel and the IP network. It performs the routing and address mapping function between the channel devices and network IP addresses.

Each network interface is assigned a unique IP address. IP addresses are only associated with the network interfaces, not with the channel interfaces. A device-to-network routing table is used to route data between the channel devices and the network interfaces. The **toGNA** address, located in the NetEx/IP protocol header, is used as the device identifier. When a NetEx message is received from the channel, the **toGNA** address is used to locate the associated IP address from the routing table. The data, along with its IP address, is then given to the IP router, which, in turn, sends the data to the correct IP destination.

Similarly, when a NetEx/IP message is received from the network, the **toGNA** address is used to locate the associated device from the routing table. The data is then given to the channel driver, which, in turn, gives the data to the correct device on the channel.

There are two ways the **toGNA-to-IP** routing entries are added to the routing table. On the first occurrence of receiving a message with a particular **toGNA** address, the Router determines there is no entry in the routing table. The Router then uses either a **Static** mapping table or **Dynamic** mapping to create an entry in the routing table. When subsequent messages destined for the same **toGNA** address are received, the Router immediately finds the entry in the routing table, and routes the message and data to the Channel or Network.

## Static Mapping

Static mapping entries are built when a **toGNA-to-IP** routing is defined during the NESiGate customization process. The static mapping entries are contained in a configuration file located on the Netfinity. This file defines the address relationship between a device's HYPERchannel GNA address and its associated IP address. Entries are placed in the file automatically when a Web browser is used to customize NESiGate. Alternatively, entries can be placed in the file manually by logging on to NESiGate as a Linux user, and creating entries in the file using the vi editor.

## Dynamic Mapping

Dynamic mapping is performed when no entries are predefined in the static routing file. Whenever a NetEx/IP message is received from the channel for which there is no entry in the routing file, a request is made to the DNS Requester component to perform a DNS lookup for a hostname entry having the following format:

NTX0000uuss

where **uu** is the network address and **ss** is the subaddress.

These addresses comprise the HYPERchannel **toGNA** address, as defined in the NetEx/IP NCT by the NETADDR (uu) and SMGDREF (ss) parameters. If the DNS lookup is successful, an entry is dynamically created in the routing table that maps the **toGNA** and **toIP** addresses. Subsequent searches of the table on behalf of new messages from the channel will find this new entry, thus avoiding additional DNS lookups.

## Control Program

The Control Program executes in user space. During initialization, it loads the NESiGate components and creates the required environment. It also provides the interface for executing Operator commands.

## DNS Requester

The DNS Requester executes in user space. It interfaces with TCP/IP DNS services to perform the DNS hostname lookups requested by the Router component. The format of the hostnames is NTX0000uuss, as described in the Router component section.

## Operator Interface

The Operator Interface component executes in user space as part of the Control Program. It provides the interface for issuing Operator Commands, as well as displaying the results. Operator commands are used to configure NESiGate, as well as to provide a display facility. Operator commands can either be issued directly by a Linux user through the command-line interface, or they can be issued through the Web Browser interface. If they are issued directly by a Linux user logged in to NESiGate, the command responses are displayed in the user's terminal session. If they are issued through the Web Browser interface, the command responses are displayed in a browser frame.

To use the command-line Operator Interface, a user must have a Linux userid and password established on NESiGate

## **Web Browser Interface**

The Apache Web server is used to provide a web-enabled configuration and display facility. Connecting NESiGate to an intranet infrastructure enables usage of a Web browser to configure, display and control the Channel, Router, IP and Control Program components.

A browser can be hosted on any system, as long as that system has connectivity to the NESiGate Apache server.

When Browser connectivity is established to NESiGate, a password must be entered on the NESiGate Web page in order to enable the user to make changes to the configuration. A password is not required for display types of commands.

## **ESCON Channel (optional)**

The ESCON channel firmware is optimized to interface with NetEx/IP and is required on NESiGate devices that have an ESCON channel. It consists of the base firmware, kernel drivers that provide the interface between the ESCON channel firmware and the channel interface, and tools used for loading firmware, displaying board-level information and configuring the ESCON devices.

Each ESCON channel interface supports up to 30 devices. Since EMIF is fully supported, the ESCON channel can be shared across multiple host images.

## **Parallel Channel (optional)**

The Parallel channel firmware is optimized to interface with NetEx/IP and is required on NESiGate devices that have a Parallel (Bus & Tag) channel. It consists of the base firmware, kernel drivers that provide the interface between the Parallel channel firmware and the channel interface, and tools used for loading firmware, displaying board-level information and configuring the Parallel channel devices.

Each Parallel channel interface supports up to 30 devices.

# NESiGate Channel Offload Adapter

This implementation is used with the channel-attached NetEx/IP Offload products for IBM OS/390 (H217IP) and Unisys (H307IP). By using the Netfinity as a channel-to-IP network gateway, H217IP and H307IP are able to communicate with other NetEx/IP hosts over standard IP-based infrastructures, such as Fast Ethernet, Gig Ethernet, or FDDI, without requiring network connectivity on the IBM or Unisys hosts.

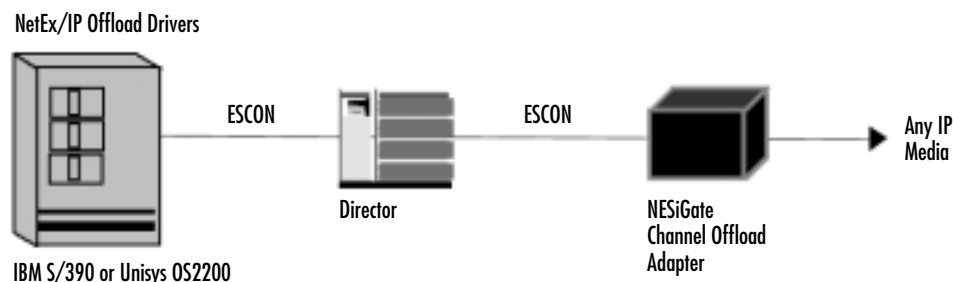
An additional benefit of using a NetEx/IP Channel Offload solution is that it may provide up to a two to five times reduction in CPU cycles, versus the more traditional host-based NetEx/IP solution. The primary reason for this reduction is because the NetEx/IP Channel Offload driver is linked with the application. This eliminates the cross-memory copying of data on the CPU and also allows the I/O to be issued to NESiGate directly from the application. There is no separate NetEx/IP subsystem to go through.

Of course, there are a few disadvantages in not having a NetEx/IP subsystem:

- NetEx/IP APR (alternate path recovery) from the host is not supported. However, it is supported from NESiGate out onto the network.
- Dedicated subchannels are required across different NetEx/IP Offload applications (i.e. no shared usage)

The decision between using host-based NetEx/IP or NetEx/IP Channel Offload basically becomes a tradeoff between more CPU cycles or reduced functionality.

The only host connectivity requirement is to have either an ESCON or Parallel channel to provide the connectivity to the Netfinity. As with all NetEx/IP Netfinity solutions, all the network connectivity is provided by the Netfinity, which allows network connections to be accessed from systems that may not natively provide support for that particular network interface.



## How It Works

A NESiGate Channel Offload configuration consists of up to two channel interfaces (ESCON and/or Parallel), as well as one or more standard network interfaces. The NESiGate software, working in conjunction with the host-based NetEx/IP drivers, provides an interface between the channel and the NetEx/IP running on NESiGate.

The primary difference between the NESiGate Channel Adapter and the NESiGate Channel Offload Adapter is that the Offload Adapter actually consists of a complete NetEx/IP host-based product running on the NESiGate platform, with host-based driver routines providing the interface to the applications.

The NESiGate Channel Offload Adapter software consists of the following components:

- Channel Interface
- Control Program
- Operator Interface
- Web Browser Interface
- NetEx/IP
- ESCON Channel (optional)
- Parallel Channel (optional)

## Channel Interface

The channel component executes in the Linux kernel and provides the interface between the Control Program and the channel firmware residing on the ESCON or Parallel channel board.

Two devices are required by each NetEx/IP Offload application — one for input and one for output. The devices are defined as a 3174 device type on IBM OS/390 systems, and as an Arbitrary Device on Unisys OS2200 systems.

## Control Program

The Control Program executes in user space. During initialization, it loads the NESiGate components and creates the required environment. It also provides the interface between the Channel component and the NetEx/IP that is running on NESiGate and processes commands issued through the Operator Interface.

## **Operator Interface**

The Operator Interface component executes in user space as part of the Control Program. It provides the interface for issuing Operator Commands, as well as displaying the results. Operator Commands are used to configure NESiGate, as well as to provide a display facility. Operator Commands can either be issued directly by a Linux user through the command-line interface, or they can be issued through the Web Browser interface. If they are issued directly by a Linux user logged in to NESiGate, the command responses are displayed in the user's terminal session. If they are issued through the Web Browser interface, the command responses are displayed in a browser frame.

To use the command line Operator Interface, a user must have a Linux userid and password established on NESiGate.

## **Web Browser Interface**

The Apache Web server is used to provide a Web-enabled configuration facility. Connecting NESiGate to an intranet infrastructure enables usage of a Web browser to configure, display and control the Channel, NetEx/IP and Control Program components.

A Browser can be hosted on any system, as long as that system has connectivity to the NESiGate Apache server.

When Browser connectivity is established to NESiGate, a password must be entered on the NESiGate Web page in order to enable the user to make changes to the configuration. A password is not required for display types of commands.

## **NetEx/IP**

The NetEx/IP component represents a fully functional host-based NetEx/IP running on the NESiGate platform. It processes NetEx/IP requests made by the host-based driver components, which are issued on behalf of the host NetEx/IP applications. A single NetEx/IP Channel Offload can be shared by NetEx/IP Channel Offload applications running on different host platforms.

Just like other host-based NetEx/IP products, the following components are contained within the NetEx/IP running on NESiGate:

- Interface
- Router
- DNS Requester

## IP Interface

NetEx/IP interfaces with TCP/IP by using UDP socket calls to deliver data to the network.

## Router

The Router function is built into NetEx/IP and performs the routing and address mapping function between the channel devices and network IP addresses.

Each network interface is assigned a unique IP address. IP addresses are only associated with the network interfaces, not with the channel interfaces. A device-to-network routing table is used to route data between the channel devices and the network interfaces. The **toGNA** address, located in the NetEx/IP protocol header, is used as the device identifier. When a NetEx/IP message is received from the channel, the **toGNA** address is used to locate the associated IP address from the routing table. The data is then passed on to UDP, which, in turn, sends the data to the correct IP destination.

Similarly, when a NetEx/IP message is received from the network, the **toGNA** address is used to locate the associated device from the routing table. The data is then given to the channel driver, which, in turn, gives the data to the correct device on the channel.

## DNS Requester

There are two ways the **toGNA**-to-IP routing entries are added to the routing table. The first way occurs during NetEx/IP initialization. At that time, a call is made to DNS Services for each of the **toGNA** addresses contained in the Network Configuration Table (NCT) that represents a path between NetEx/IP and each remote NetEx/IP.

The call to DNS is to obtain the IP address for a hostname having the following format:

NTX0000uu**ss**

where **uu** is the network address and **ss** is the subaddress.

These addresses comprise the HYPERchannel **toGNA** address, as defined in the NetEx/IP NCT by the NETADDR (uu) and SMGDREF (ss) parameters. If the DNS lookup is successful, an entry is dynamically created in the routing table that maps the **toGNA** and **toIP** addresses. Subsequent searches of the table on behalf of

NetEx/IP messages from either the channel or the network will find this entry to get the corresponding IP address.

The second way a **toGNA-to-IP** entry can be added to the routing table is by using a NetEx/IP command to define a static route definition. This NetEx/IP command can be issued by using either the NESiGate Operator Interface or the Web Browser interface.

### **ESCON Channel (optional)**

The ESCON channel firmware is optimized to interface with the NetEx/IP host-driver and is required on NESiGate devices that have an ESCON channel. It consists of the base firmware, kernel drivers that provide the interface between the ESCON channel firmware and the channel interface, and tools used for loading firmware, displaying board-level information and configuring the ESCON devices.

Each ESCON channel interface supports up to 30 devices. Since EMIF is fully supported, the ESCON channel can be shared across multiple host images.

### **Parallel Channel (optional)**

The Parallel channel firmware is optimized to interface with the NetEx/IP host-driver and is required on NESiGate devices that have a Parallel (Bus & Tag) channel. It consists of the base firmware, kernel drivers that provide the interface between the Parallel channel firmware and the channel interface, and tools used for loading firmware, displaying board-level information and configuring the Parallel channel devices.

Each Parallel channel interface supports up to 30 devices.

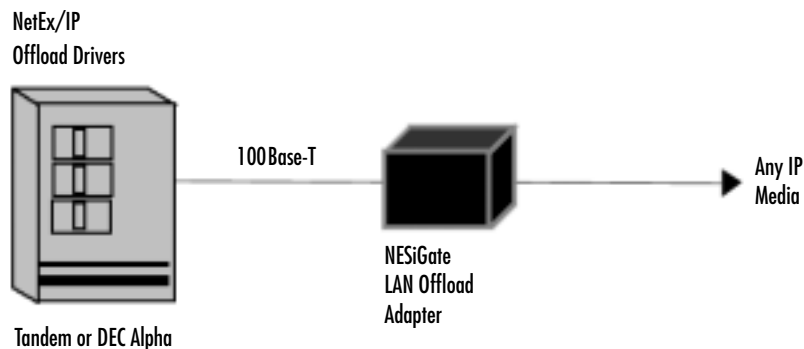
### **NESiGate LAN Offload Adapter**

This implementation is used with the LAN-attached NetEx/IP Offload products for Tandem (H367IP) and DEC Alpha (H267IP). By using the Netfinity as a LAN-to-IP network gateway, H267IP and H367IP are able to communicate with other NetEx/IP hosts over standard IP-based infrastructures, such as Fast Ethernet, Gig Ethernet, or FDDI, without requiring anything other than an Ethernet connection on the DEC or Tandem hosts.

As is the case with NetEx/IP Channel Offload, the NetEx/IP LAN Offload driver is also linked with the application. The application passes NetEx/IP requests directly to

NetEx/IP LAN Offload using TCP/IP. NetEx/IP LAN Offload processes the NetEx/IP requests, and manages and maintains communication with NetEx/IP remote hosts.

The only host connectivity requirement is to use a LAN interface to provide the connectivity to the Netfinity. As with all NetEx/IP Netfinity solutions, the network connectivity is provided by the Netfinity, which allows network connections to be accessed on systems that may not natively provide support for that particular network interface.



## How It Works

A NESiGate LAN Offload configuration consists of at least one network interface for connectivity to the host, as well as one or more standard network interfaces for connectivity to the NetEx/IP network. The NESiGate software works in conjunction with the host-based NetEx/IP drivers.

The NESiGate LAN Offload Adapter software consists of the following components:

- Control Program
- Operator Interface
- Web Browser Interface
- NetEx/IP

## Control Program

The Control Program executes in user space. During initialization, it loads the NESiGate components and creates the required environment. It also provides the interface between TCP/IP and NetEx/IP for the purpose of sending and receiving

NetEx/IP requests to and from the host. It processes commands issued through the Operator Interface.

## **Operator Interface**

The Operator Interface component executes in user space as part of the Control Program. It provides the interface for issuing Operator Commands, as well as displaying the results. Operator Commands are used to configure NESiGate, as well as to provide a display facility. Operator Commands can either be issued directly by a Linux user through the command line interface, or they can be issued through the Web Browser interface. If they are issued directly by a Linux user logged in to NESiGate, the command responses are displayed in the user's terminal session. If they are issued through the Web Browser interface, the command responses are displayed in a browser frame.

To use the command line Operator Interface, a user must have a Linux userid and password established on NESiGate.

## **Web Browser Interface**

The Apache Web server is used to provide a Web-enabled configuration facility. Connecting NESiGate to an intranet infrastructure enables usage of a Web browser to configure, display and control the NetEx/IP and Control Program components.

A browser can be hosted on any system, as long as that system has connectivity to the NESiGate Apache server.

When browser connectivity is established to NESiGate, a password must be entered on the NESiGate Web page in order to enable the user to make changes to the configuration. A password is not required for display types of commands.

## **NetEx/IP**

The NetEx/IP component represents a fully functional host-based NetEx/IP running on the NESiGate platform. It processes NetEx/IP requests made by the Host based driver components, which are issued on behalf of the host NetEx/IP applications. A single NetEx/IP LAN Offload can be shared by NetEx/IP LAN Offload applications running on different host platforms.

Just like other host-based NetEx/IP products, the following components are contained within the NetEx/IP running on NESiGate:

- IP Interface
- Router
- DNS Requester

## IP Interface

NetEx/IP interfaces with TCP/IP by using UDP socket calls to deliver data to the network.

## Router

The Router function is built into NetEx/IP and performs the routing and address mapping function between NetEx/IP devices and network IP addresses.

Each network interface is assigned a unique IP address. However, IP addresses are only associated with the network interfaces, not with the NetEx/IP devices. A device-to-network routing table is used to route data to the appropriate IP address. The **toGNA** address, located in the NetEx/IP protocol header, is used as the device identifier. When a NetEx/IP message is received from the host, the **toGNA** address is used to locate the associated IP address from the routing table. The data is then passed on to UDP, which, in turn, sends the data to the correct IP destination.

Similarly, when a NetEx/IP message is received from the network, the **toGNA** address is used to locate the associated device from the routing table. The data is then given to the Control Program, which, in turn, returns the data to the host.

## DNS Requester

There are two ways the **toGNA**-to-IP routing entries are added to the routing table. The first way occurs during NetEx/IP initialization. At that time, a call is made to DNS Services for each of the **toGNA** addresses contained in the Network Configuration Table (NCT) that represents a path between NetEx/IP and each remote NetEx/IP.

The call to DNS is to obtain the IP address for a hostname having the following format:

NTX0000uuss

where **uu** is the network address and **ss** is the subaddress.

These addresses comprise the **HYPERchannel toGNA** address, as defined in the NetEx/IP NCT by the NETADDR (uu) and SMGDREF (ss) parameters. If the DNS lookup is successful, an entry is dynamically created in the routing table that maps the **toGNA** and **toIP** addresses. Subsequent searches of the table on behalf of NetEx/IP messages from either the host or the network will find this entry to get the corresponding IP address.

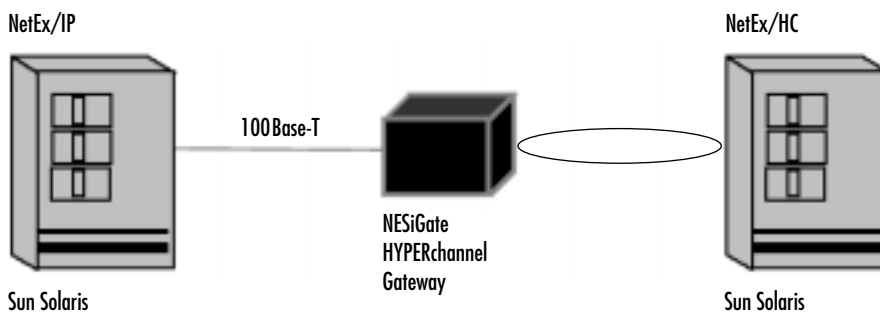
The second way a **toGNA-to-IP** entry can be added to the routing table is by using a NetEx/IP command to define a static route definition. This NetEx/IP command can be issued by using either the NESiGate Operator Interface or the Web Browser interface.

## NESiGate HYPERchannel Gateway Adapter

This implementation is used to interconnect a NetEx/IP network with a legacy NetEx/HC network. The HYPERchannel side of this NESiGate device is an FDDI network interface card that can connect on an FDDI ring to any other NetEx HYPERchannel devices — either DX(E) or direct FDDI NIC's on NetEx hosts. By using the Netfinity as a HYPERchannel-to-IP network gateway, NetEx/HC applications are allowed to participate in a NetEx/IP network and to communicate with NetEx/IP applications. Communications on the NetEx/IP side of this NESiGate device can take place over any standard IP-based infrastructures, such as Fast Ethernet, Gig Ethernet or FDDI.

There is no architected limit to the number of NetEx/HC hosts that can be connected to a single HC gateway device.

As with all NetEx/IP NESiGate solutions, the IP network connectivity is provided by the Netfinity, which allows network connections to be accessed from systems that may not natively provide support for that particular network interface.



## How It Works

A NESiGate HYPERchannel to IP Gateway Adapter consists of one FDDI network interface card for connectivity to the HYPERchannel FDDI ring, as well as one or more standard network interface cards for connectivity to the NetEx/IP network. NESiGate contains special routing software to properly route and format data between IP and HYPERchannel.

On the HYPERchannel side of the network, this NESiGate device essentially acts as a proxy for the remote NetEx/IP hosts. The NESiGate device must be configured during the customization process to accept the **toGNA** addresses of the remote NetEx/IP hosts. This forces all FDDI packets that are destined to the remote NetEx/IP to be picked up by the Gateway device. After the Gateway gets the HYPERchannel packets, it strips off the FDDI header, does a DNS lookup (if needed) for **NTX0000uuss** (where **uuss** identifies the **toGNA** address of the remote NetEx/IP host), and sends the data to the IP network by using a UDP socket call.

On the IP side of the network, this NESiGate device acts as a proxy for the remote NetEx/HC hosts. The NESiGate device must receive all IP packets destined for a remote NetEx/HC host. This is done by defining all **NTX0000uuss** hostnames (where **uuss** identifies the **toGNA** address of the remote NetEx/HC hosts) to be the IP address of a network card on the Gateway device. After the Gateway device gets the IP packets, it strips off the IP header, encapsulates the HYPERchannel data in an FDDI packet and sends the data on the FDDI network.

A NESiGate HYPERchannel-to-IP Gateway Adapter consists of the following components:

- HCG FDDI Interface
- IP Interface
- Router
- Control Program
- DNS Requester
- Operator Interface
- Web Browser Interface

## **HCG FDDI Interface**

This module is loaded into the Linux kernel during NESiGate initialization. It provides the interface between the Router component and the HYPERchannel FDDI NetEx/HC network. For data going out on the FDDI network, this component encapsulates the data into FDDI packets. For data coming in from the FDDI network, this component removes the FDDI packet headers.

## **IP Interface**

This module is loaded into the Linux kernel during NESiGate initialization. It provides the interface between the Router component and the TCP/IP stack. It interfaces with TCP/IP by using a UDP socket interface.

## **Router**

This module is loaded into the Linux kernel during NESiGate initialization. It provides the interface between the HCG FDDI network and the IP network. It performs the routing and address mapping function between the HYPERchannel network and the IP network.

## **Control Program**

The Control Program executes in user space. During initialization, it loads the NESiGate components and creates the required environment. It also provides the interface for executing Operator Commands.

## **DNS Requester**

This program executes in user space. It provides the support for performing DNS name lookups, if required, during Router processing.

## **Operator Interface**

The Operator Interface component executes in user space as part of the Control Program. It provides the interface for issuing Operator Commands, as well as displaying the results. Operator Commands are used to configure NESiGate, as well as to provide a display facility. Operator Commands can either be issued directly by a Linux user through the command line interface, or they can be issued through the Web Browser interface. If they are issued directly by a Linux user logged in to NESiGate, the command responses are displayed in the user's terminal session. If

they are issued through the Web Browser interface, the command responses are displayed in a browser frame.

To use the command line Operator Interface, a user must have a Linux userid and password established on NESiGate.

## **Web Browser Interface**

The Apache Web server is used to provide a Web-enabled configuration facility. Connecting NESiGate to an intranet infrastructure enables usage of a Web browser to configure, display and control the NetEx/IP and Control Program components.

A browser can be hosted on any system, as long as that system has connectivity to the NESiGate Apache server.

When Browser connectivity is established to NESiGate, a password must be entered on the NESiGate Web page in order to enable the user to make changes to the configuration. A password is not required for display types of commands.