Bridge and Extended LAN

digital™

Reference

Order Number: EK-DEBAM-HR-003

NOTICE – Class 1 Laser Device:
The lasers in this equipment are Class 1 devices, compliant with CDRH Rules 21, CFR Subchapter J, Part 1040.10, at date of manufacture. Class 1 laser devices are not considered to be hazardous.

NOTICE – Class A Computing Device:
This equipment generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such radio frequency interference when operated in a commercial environment. Operation of this equipment in a residential area may cause interference; in which case, measures taken to correct the interference are at the user's expense.

CAUTION

The people who install the cabling system described in this guide should be familiar with local building codes, fire codes, and any other applicable codes or regulations. The manufacturers or their distributors and agents will not be responsible for damage due to improperly installed cabling, neglect, misuse, or improper connection of devices to the cabling system.

Bridge and Extended LAN

Reference

September 1991

This guide describes how bridges are used to create extended local area networks (LANs). The descriptions include the use of bridges in extended LAN configurations, information on LAN interconnections, overall bridge operation, spanning tree, bridge management, and solving bridge-related problems in a network.

Supersession/Update Information: This is a revised manual.

Order Number: EK-DEBAM-HR-003

EK–DEBAM-HR-003
SEPTEMBER 1991

This manual was produced by Telecommunications and Networks Publications.

Contents

2 Bridge Operation

3 The Spanning Tree

4 Extended LAN and Bridge Management

5 Bridge and Extended LAN Performance

6 Configuration

7 Network Troubleshooting Methodology

Figures

Tables

Preface

As a local area network (LAN) grows, it can exceed the design parameters of an individual LAN. Restrictions (such as physical extent, number of stations, performance, and media) can be alleviated by the interconnection of multiple LANs. All single LAN topologies have limitations. Bridges address these limitations and provide a good solution as the interconnecting device between the multiple LANs. The user perceives a logical LAN connection to all devices on the extended LAN, while the physical LAN constraints are met by each LAN.

Objectives of This Guide

This guide describes how to use bridges in extended LAN configurations, including information on LAN interconnections, overall bridge operation, extended LAN topologies, bridge management, and solving bridge-related problems in a network.

Intended Audience

This guide is intended to provide an understanding of bridges and extended LANs for use by network managers and other personnel whose tasks are related to setting up, managing, and troubleshooting LANs and extended LANs.

Structure of This Document

This guide has seven chapters and three appendixes, as follows:

Chapter 1	Provides an overview of bridges and introduces a bridge model. It includes information on extended LANs and types of bridges, and discusses bridge features and services.
Chapter 2	Provides a detailed conceptual overview of bridge operations, using the bridge model introduced in Chapter 1.
Chapter 3	Provides a conceptual overview of the spanning tree algorithm and a detailed description of its operation.
Chapter 4	Provides an overview of bridge management and, using the bridge model, discusses management capabilities of the model.
Chapter 5	Discusses bridge and extended LAN performance from a user's point of view.
Chapter 6	Discusses physical and logical topologies, including unadvised topologies and how to modify the network to achieve maximum benefits.
Chapter 7	Provides an overview of network troubleshooting tools and methodologies.
Appendix A	Lists the various types of bridges that are available from Digital and compares the features for each model.
Appendix B	Briefly describes the DECbridge 90 workgroup bridge.
Appendix C	Provides a list of related product documentation orderable from Digital.

The postage-paid Reader's Comments form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

Related Documents

Refer to Appendix C for a list of additional documents that provide related information about Digital Equipment Corporation's bridges and networks. Ordering information is provided at the back of this guide.

1

Introduction

This chapter provides an overview of bridges, develops a general bridge model, and uses the model description to provide the reader with basic knowledge of the operation of bridges. Although the bridge model is general, it can help network implementers understand the operation of different bridge implementations and the services provided by bridges.

1.1 Types of Bridges Discussed in This Guide

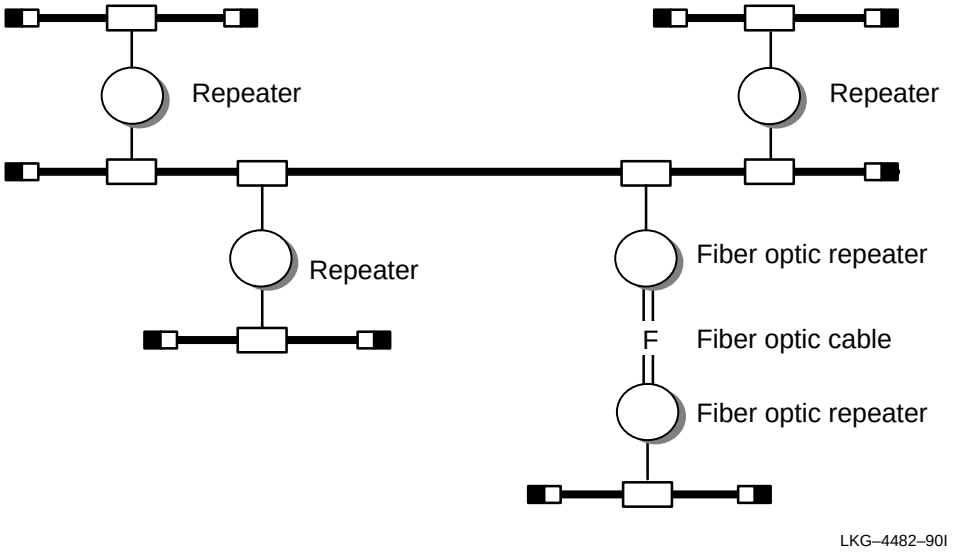
Bridges are designed with a wide variety of features, data link types, performance, and uses. This guide does not detail all available bridges, but, by understanding the basic concepts and implementation tradeoffs, network implementers should be able to select a product that is appropriate to their needs for LAN interconnectivity.

Appendix A lists the bridges that are available from Digital Equipment Corporation and compares the available features for each model.

1.2 Single Local Area Networks

A local area network (LAN) is generally a privately owned data communications system that offers a shared high-speed communications channel optimized for connecting information-processing equipment (see Figure 1–1). A LAN usually serves a trusted work group in a section of a building, an entire building, or a cluster of buildings.

Figure 1–1: Single LAN Example (Ethernet)



Single LANs typically consist of media segments connected by repeaters in what is logically considered a single transmission bus. Any data present on a media segment is transmitted throughout the entire LAN.

All LANs have similar basic limitations. A shared transmission medium requires unanimous agreement by all LAN users on certain basic parameters, such as transmission speed, error checking, packet sizes, and the number of connections. The basic standardized parameters lead to a set of limitations for any given LAN technology.

Some general limitations are:

- Number of stations
- Throughput (speed)
- Length
- Topology (ring, bus, tree)
- Access delay
- Reliability and availability
- Manageability and maintainability

The basic LAN limitations apply to all LAN technologies. In general, the bandwidth, number of stations, and length of any single LAN will be limited by the technology used to create it. These restrictions must be adhered to for the LAN to operate as it was designed.

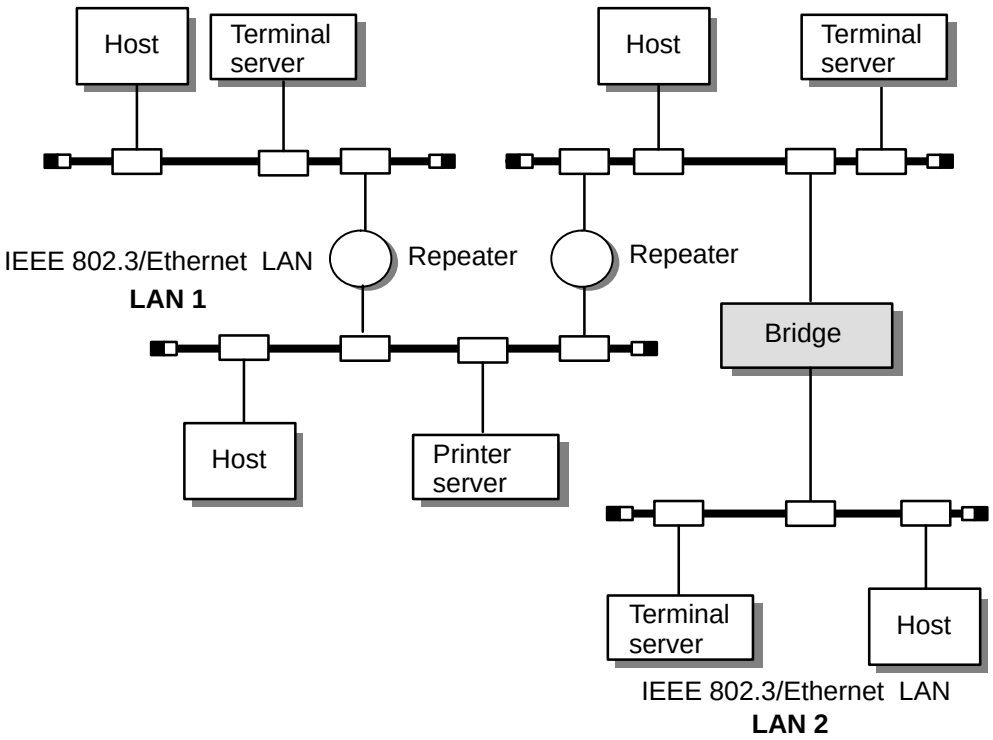
A simple, cost-effective way to overcome the limitations inherent in a single-LAN topology is to use bridges to interconnect LANs to form extended LANs. This use of bridges preserves the basic LAN services while extending those services beyond the limitations of a single-LAN topology. The following sections describe how bridges form extended LANs to provide this solution.

1.3 The Extended LAN Concept

An **extended LAN** is a collection of local area networks interconnected by protocol-independent store-and-forward devices (bridges). Using the bridge to create a LAN-to-LAN interconnection allows each station on the attached LANs to communicate with all stations on both LANs as if those stations were on the same LAN.

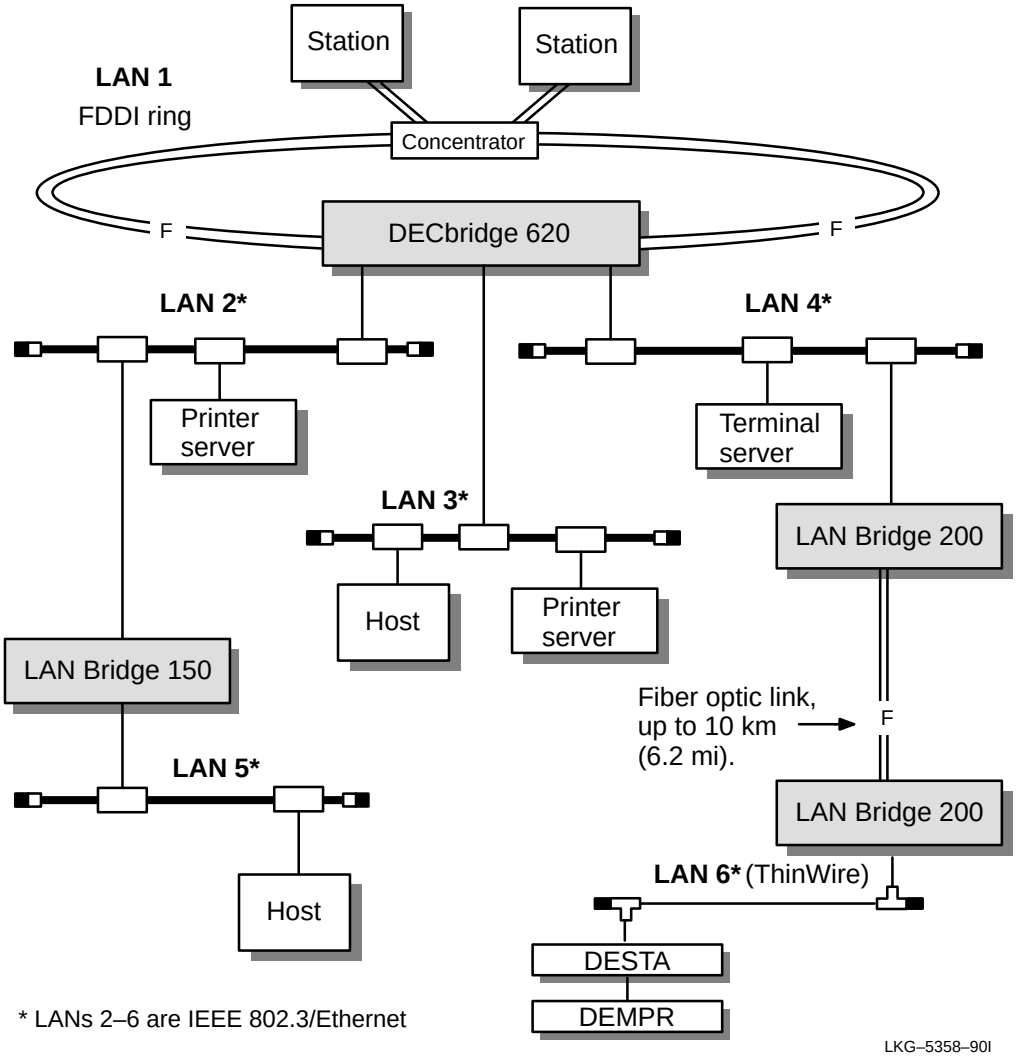
Bridges are the connecting devices between local area networks in an extended LAN architecture. Acting as data link relays between *similar or dissimilar* LAN types, bridges allow the creation of extended LANs (see Figure 1–2 and Figure 1–3).

Figure 1–2: Extended LAN Example (Similar LAN Types)



LKG–4483–90I

Figure 1–3: Extended LAN Example (Dissimilar LAN Types)



Bridges dynamically learn the station addresses of nodes for each station within the extended LAN. This enables the bridge to forward frames selectively, based on the destination address of the frame. When a frame is received on one port, it examines the frame's address to determine whether it should be passed to the other port. This concept also applies to **multiport bridges** (bridges with more than two ports).

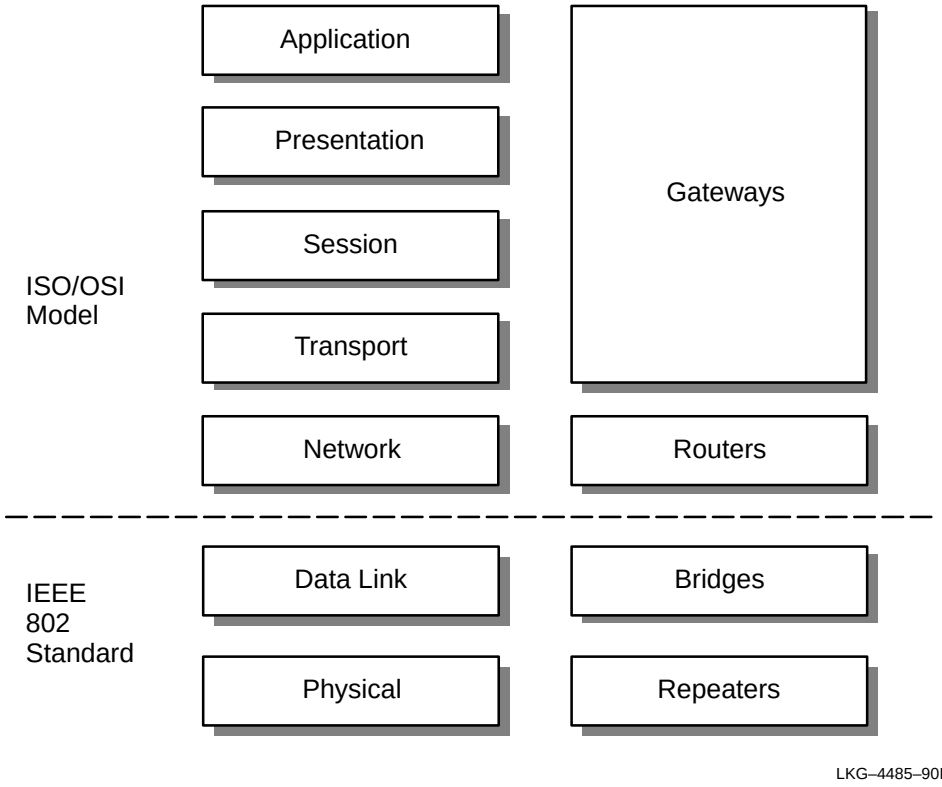
Bridges create extended LANs that have the following advantages:

- **More stations** — Each LAN can still support its maximum number of stations. The bridge counts as one station on each LAN it is attached to.
- **Larger network** — Each LAN can still support the maximum length for the given technology. The extended LAN can be much longer since a frame can be forwarded by several bridges across multiple LANs.
- **Reduced traffic** — The bridge does not forward local traffic. Thus, if a large LAN is partitioned into several smaller LANs by bridges, the total traffic is reduced by the amount of traffic that is localized to each of the smaller LANs.
- **Interconnection of different LANs** — Bridges allow the interconnection of LANs that can employ dissimilar media access control (MAC) or different physical media. This allows network designers to use the type of LAN that is best suited to the network design and environmental requirements. This concept is often expanded to include links into the wide area network (WAN) environment.
- **LAN availability** — The spanning tree algorithm (refer to Chapter 3) allows redundant paths to be physically configured. This can improve the availability of services on the extended LAN.
- **Management availability** — As a LAN interconnection device, the bridge is ideally situated to aid in LAN and extended LAN management. LAN utilization, error monitoring, and bridge throughput are all valuable capabilities that can easily be provided by a bridge management entity within the bridge to either a local or remote management station. Other important capabilities are:
  - Error isolation
  - LAN fault partitioning
  - Statistics gathering
  - Topology information
  - Access control
  - Node locations

1.4 Bridge Concept

A **bridge** is a protocol-independent device that connects local area networks, providing logical data link services for all stations on those attached LANs. Conceptually, the bridge can also be thought of as a data link relay that forwards frames to or from the MAC sublayers and the physical channels of the attached LANs, thus providing data link connectivity between them (see Figure 1–4). The actual implementation of these services often leads to a blend of several layers of the International Organization for Standardization (ISO), Open Systems Interconnection (OSI) model. Sections 1.4.1 through 1.4.4 describe the network devices in more detail.

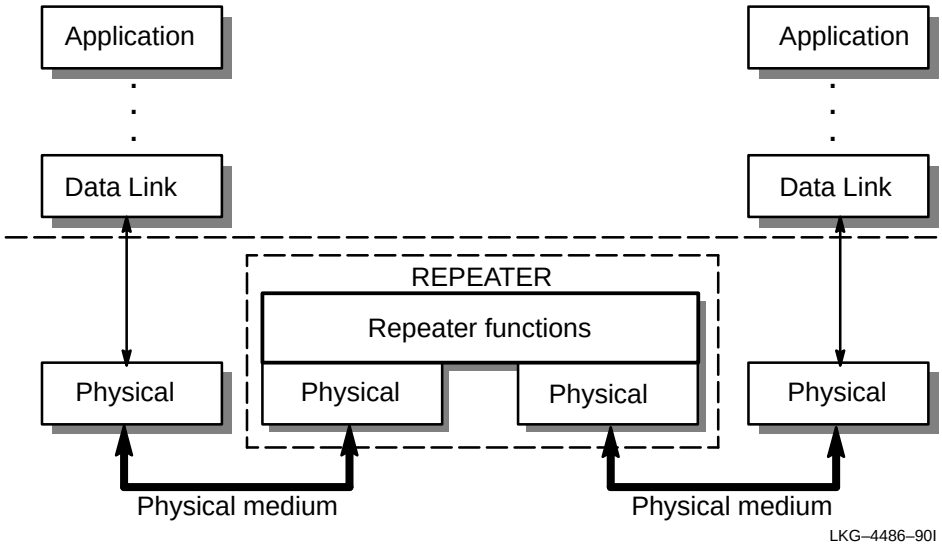
Figure 1–4: ISO/OSI Reference Model



1.4.1 Repeaters

As shown in Figure 1–5, repeaters operate on the Physical layer of the ISO/OSI model and are transparent to the upper layers. Repeaters do not use addressing. They provide a bit store-and-forward function while amplifying and restoring timing margins to the packet bit stream.

Figure 1–5: Repeaters and the Physical Layer

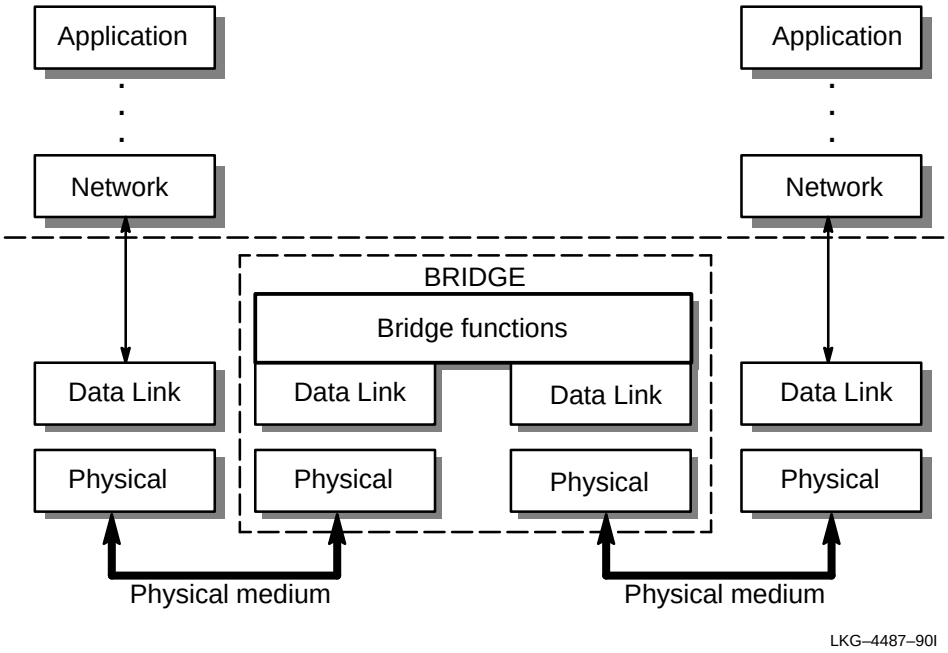


Repeaters are also useful for overcoming the limitations of physical media. All cables have attenuation, introduce noise, and are subject to faults. Repeaters can overcome these types of limitations while providing topological flexibility, such as allowing a bus topology to become a branching tree topology.

1.4.2 Bridges

As shown in Figure 1–6, bridges operate on the Data Link layer and are transparent to layers at and above the Network layer in the ISO/OSI model. Bridges interpret media access control (MAC) addresses of the network and forward data link frames.

Figure 1–6: Bridges and the Data Link Layer

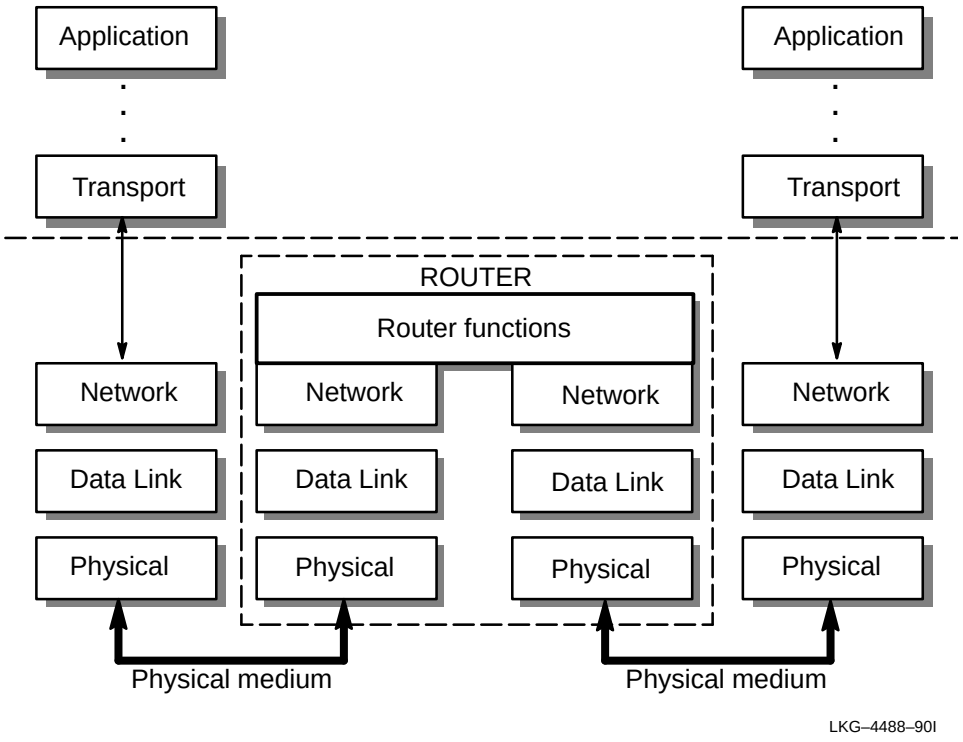


Although bridges operate one layer lower than routers (just below the Network layer), the gap between the two types of devices sometimes overlap as more intelligent bridges actually look beyond the data link frame into the Network layer frame. This functionality allows the bridge to filter traffic based on protocol type or even particular data bit patterns, while remaining transparent to upper layers. Digital's LAN Bridge 200 and DECbridge 500/600 series are designed this way.

1.4.3 Routers

As shown in Figure 1–7, routers operate on the Network layer of the ISO/OSI model and are specific to a given routing protocol. Network addressing is used at this layer.

Figure 1–7: Routers and the Network Layer

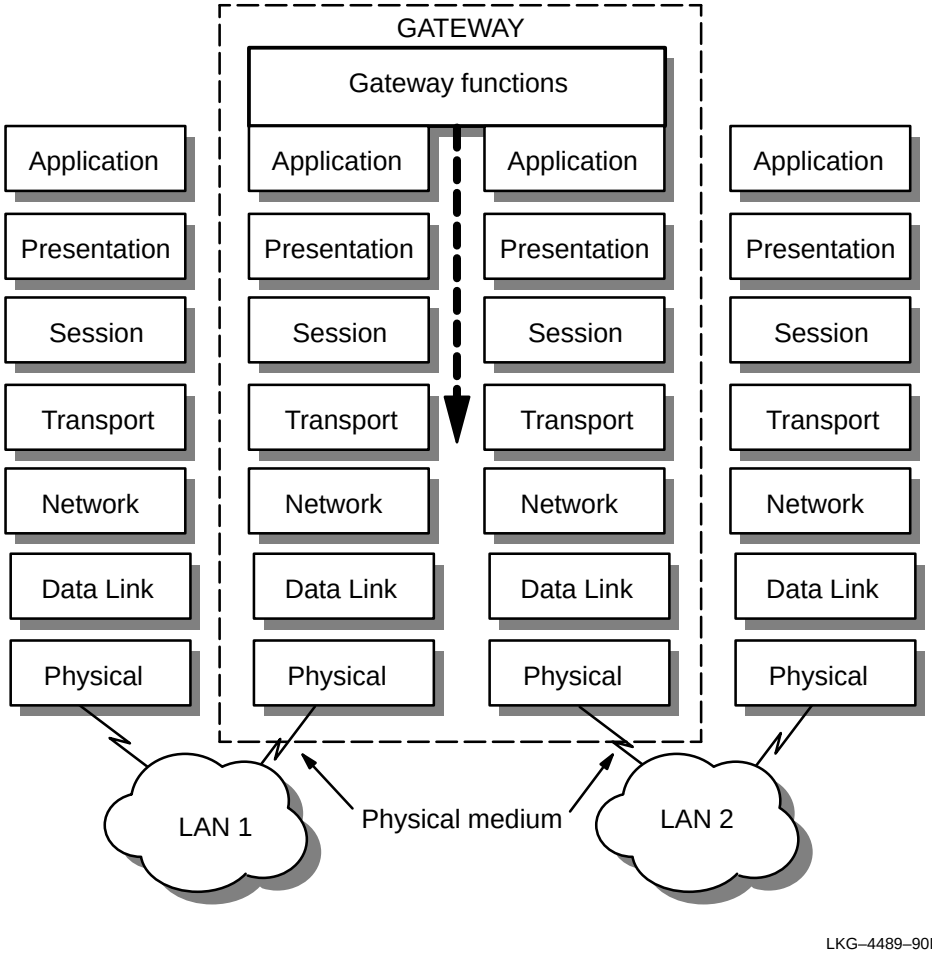


Routers operate on protocols (such as DECnet, Transmission Control Protocol/Internet Protocol (TCP/IP), XNS, or other standard protocols) and only pass the traffic of the protocol types that it is designed to support.

1.4.4 Gateways

As shown in Figure 1–8, gateways refer to the class of devices operating above the Network layer of the ISO/OSI model.

Figure 1–8: Gateways and the Application Layer

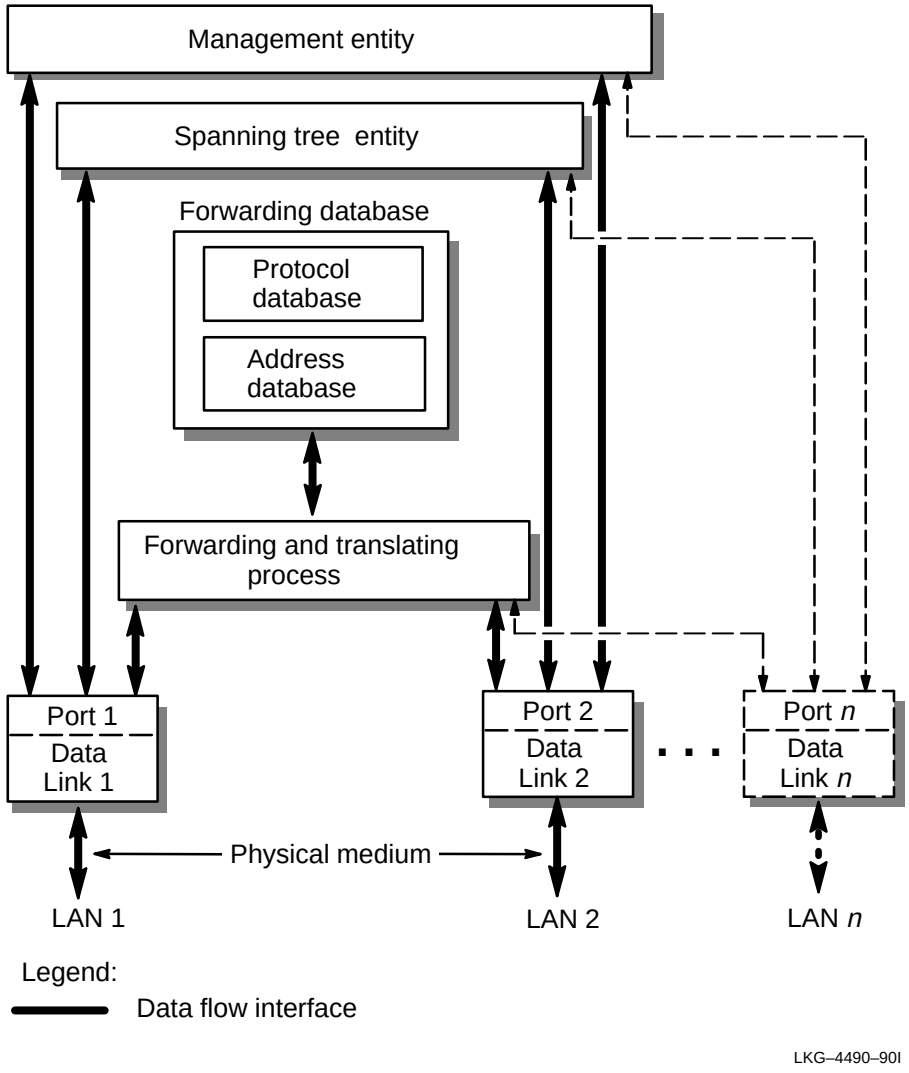


Gateways provide different types of translation functions, including network address, coding, access type, or applications (such as Mail gateways and DECnet/SNA gateways).

1.5 Bridge Model

The bridge model (see Figure 1–9) represents the major functional blocks of a bridge.

Figure 1–9: Bridge Data Path Functional Model



Each of the functional blocks represented in the model is briefly described in the following sections. A detailed description of the functional blocks, including their interactions, is provided in Chapter 2 through the use of an expanded bridge model.

1.5.1 Port Interfaces

The ports provide transmit and receive services to the forwarding and translating process module and to the management entity. The bridge architecture allows for an unlimited number of ports.

Port interfaces on the bridge provide the data link and physical access for the bridge to each attached LAN. Bridge ports are specific to the type of attached LAN and contain information about the characteristics and status of the Physical layer, data link, and internal interfaces. The ports have two service interfaces: an interface that provides all (error-free) frames to the forwarding and translating process module, and an interface that provides services to the bridge's spanning tree and management entity. Either service interface can receive or transmit frames to the attached LAN.

1.5.2 Forwarding and Translating Process Module

The forwarding and translating process module receives frames from the port interfaces and decides whether to forward or discard the frames based on information contained in the forwarding database. Frames destined to be forwarded need to be translated when dissimilar type networks are connected by bridges. The DECbridge 500/600 series is an example of bridges with this functionality. They modify the protocol header of the forwarded packet to make it compliant with the network it is being forwarded to, and recalculate the Cyclic Redundancy Check (CRC). In addition, if the size of an Internet Protocol (IP) frame is larger than the frame size allowed by the bridge's outbound port, the DECbridge 500/600 series *fragments* the frame into several smaller packets to accommodate the requirements of the recipient LAN (refer to Section 2.1.3 for more information).

1.5.3 Forwarding Database

The forwarding database consists of the address database and the protocol database. The bridge forwards data frames based on the destination addresses, source addresses, and protocol information contained in these associated databases.

1.5.4 Address Database

The address database contains a list of known source addresses in the extended LAN. Each address contains a port number that the bridge associates with the station's source address. Special addresses (such as the bridge's own address) are permanently placed in the table. A user with management software can set others in the table if desired. Other status may be associated with each address and will be discussed later.

1.5.5 Protocol Database

The protocol database is set up only by the management entity and may not be present in all bridges. It contains a list of protocols and the type of service that they would receive. The forwarding and translating process module uses the protocol database to decide whether the protocol contained in the frame should be forwarded. A frame is only forwarded if it passes all filters that were enabled in the forwarding and translating process module. More information on protocol filtering is provided throughout this manual (refer to the index).

1.5.6 Management Module

The management module executes all bridge-related management functions and, in general, has control and status access to the entire bridge. This module can receive and respond to network management requests from management stations. For simplicity in the model, the management module can also be thought of as containing the spanning tree entity.

1.5.6.1 Management Entity

There can be one or more management entities within the management module in the bridge. Although management entities are not required, most bridges provide at least one entity that enables remote users to gather statistics and state information from the bridge. This can be accomplished from across the attached LANs or through a dedicated external interface.

The status LEDs and switches on a bridge are a dedicated local management entity that is also contained in the management module.

1.5.6.2 Spanning Tree Entity

The spanning tree entity contains the implementation of the spanning tree algorithm that determines the state of the ports. By controlling which ports are enabled on the bridge, the spanning tree entity can resolve a general mesh topology of any configuration of bridges and LANs into a fully connected spanning tree. A spanning tree ensures full connectivity in the extended LAN while avoiding data link loops.

1.6 Services Provided by Bridges

Services provided by bridges include:

- **Store-and-forward capability** — Bridges receive, check, and transmit frames to other LANs, enabling the configuration of extended LANs.
- **Frame filtering based on address** — Using the address database and the source and destination address from incoming frames, the forwarding and translating process module isolates the traffic that *should not be allowed* on other LANs. This action reduces the total data traffic on an extended LAN by not forwarding the packets that have local destination addresses or packets that are not allowed to be forwarded. This increases bandwidth efficiency.
- **Data Link layer relay** — Operation at this layer makes the bridge transparent to the protocols that use the LAN connectivity service. This protocol transparency is a key factor in the extended LAN service.
- **Dynamic address learning** — The forwarding and translating process module automatically adds new source addresses to the address database while the bridge is operating. This *reverse learning* of the address and port association allows automatic configuration of the network without prior downline loading of configuration data to the bridge. Note that the address learning is protocol and management entity independent. This process allows a *plug and play* philosophy to be applied to extended LAN growth.
- **Port independence** — An implementation can be built into bridges for any compatible data link. When data links are not compatible, services can only be offered that are compatible to both data links used by the forwarding and translating process module.

- **Automatic backup (using redundant bridges)** — The spanning tree entity, with its Mesh Topology Resolution algorithm, can be used to provide redundancy and increased availability in the extended LAN environment.

- **Translation** — Digital's DECbridge 500/600 series units are IEEE 802.3/Ethernet-to-FDDI translating bridges (as opposed to encapsulating type bridges, which require a de-encapsulation device at the receiving end). Translating type bridges guarantee interoperability and transparency to upper level protocols, by creating packets that are standard on all LANs.

The following services are either not specified in the IEEE 802.1d standard or are optional:

- **Software available for enhanced management ability** — All bridges can run optional management entity code to allow remote access. This management capability allows the setting of addresses and protocols and the examination of the entire address database for determining node locations.

- **Protocol filtering** — Using network management software, a user can specify protocols that the bridge will filter regardless of the source or destination address.

- **Special links** — Implementations may embed special link technologies, such as microwave transmission, to provide connectivity across special environments.

- **Downline load capabilities** — Digital's DECbridge 500/600 series supports a network device upgrade utility that lets you upgrade the bridge firmware from either a VMS-based or ULTRIX-based host system. The utility is supplied with the upgrade software.

- **Fragmentation** — 802.3/Ethernet LANs do not have the capability to handle large frame sizes created by FDDI LANs (up to 4500 bytes in length). To ensure that 802.3/Ethernet frame size restrictions are not violated when frames cross the FDDI-to-802.3/Ethernet bridge, Digital's DECbridge 500/600 series performs a process called *fragmentation* on large Internet Protocol (IP) frames. The bridge performs the fragmentation process on frames that are larger than the frame size allowed by the outbound port. Refer to Section 2.1.3 for more information about the fragmentation feature.

1.7 Other Capabilities

The **LAN Traffic Monitor (LTM)** offered by Digital operates on the LAN Bridge 100 and LAN Bridge 150 hardware base. While configured as an LTM, these devices relinquish their bridge function and operate only as LTM *listeners*. In this mode of operation, the listener gathers network information for statistical analysis by a remote station. A standby LAN Bridge 100 or LAN Bridge 150 can be enabled to become an LTM. However, while in this mode of operation, the device no longer participates in the spanning tree computation process.

This guide does not detail LTM capabilities. For more information about Digital's LAN Traffic Monitor, refer to the *LAN Traffic Monitor User's Guide*.

Refer to Appendix A for a list of features offered with Digital's bridge family.

2

Bridge Operation

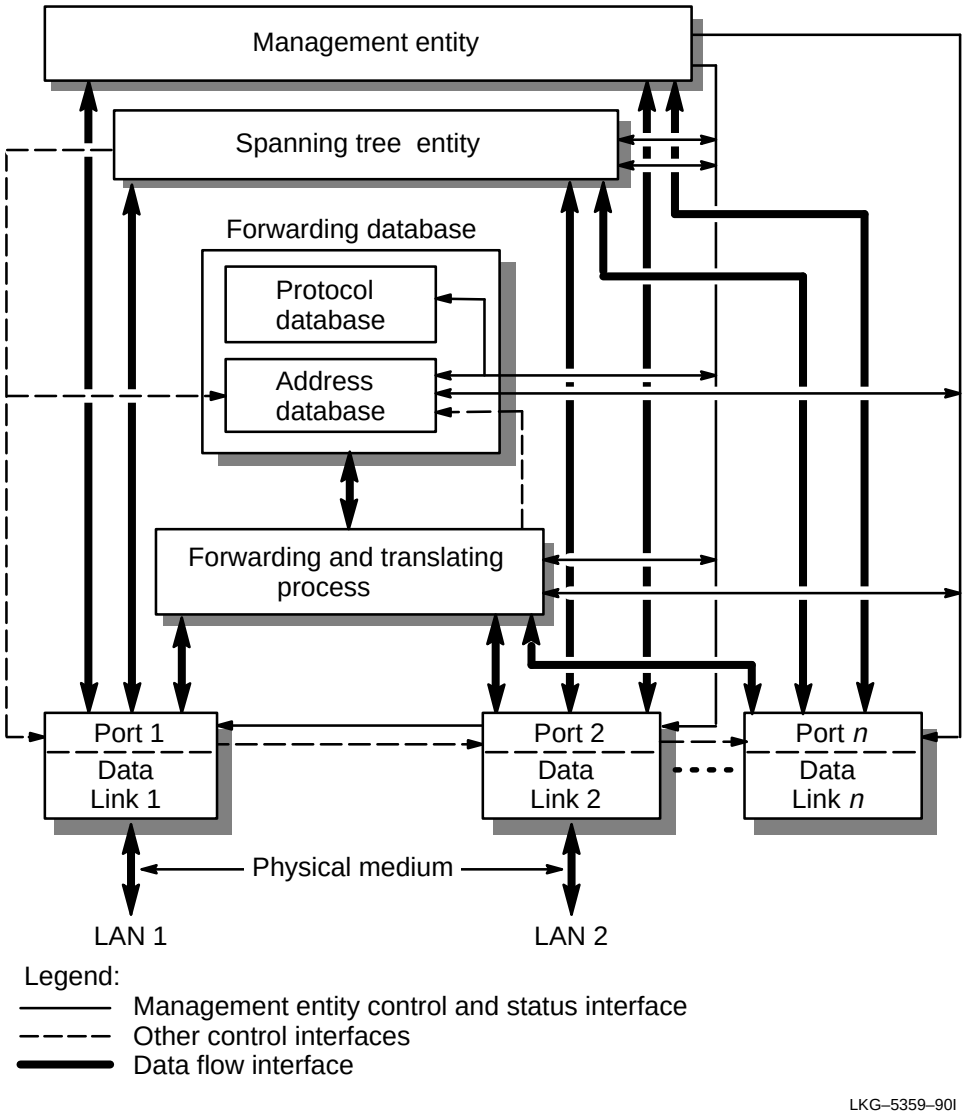
This chapter provides a detailed description of bridge operation through the use of the bridge model introduced in Chapter 1. The model has been expanded to show the internal interfaces that connect the individual function modules of the bridge. Detailed descriptions of the function modules are also provided in this chapter.

This chapter also explains and describes the different states that exist within a bridge. Bridges use these states to automatically control data flow within the bridge. The states can also be affected by the intervention of bridge management software and from the operation of the spanning tree algorithm. Understanding these states and their relationships is necessary in defining the operations of the components that make up the bridge model.

2.1 Bridge Model

Figure 2–1 shows the various interfaces associated with the individual function modules of the bridge model. Although the model shows a two-port bridge, the descriptions also apply to multiport bridges (bridges with more than two ports).

Figure 2–1: Bridge Control and Data Path Functional Model



2.1.1 Bridge Model Internal Interfaces

As shown in the bridge model (see Figure 2–1), there are three categories of internal interfaces:

- **Management entity control and status interfaces** — The management entity has a control and status interface into each of the function modules of the bridge. The management entity performs operations through these interfaces. The operations include:
  - Initialization (and self-test)
  - Gathering status and statistics
  - Setting state
  - Setting information or characteristics
  - Enabling or disabling a function

  Notice that because the only control path to the protocol database is from the management entity (see Figure 2–1), only the management entity can add information to the protocol database.

- **Other control interfaces** — The spanning tree entity has a control interface to each port. In many ways, this entity is similar to the management interface in the bridge, but is limited to controlling the port's data path to the forwarding and translating process module.

  The control interface from the forwarding and translating process module to the address database allows the forwarding and translating process to add new addresses to the address database. Addresses that are currently in the address database may have their status updated also.

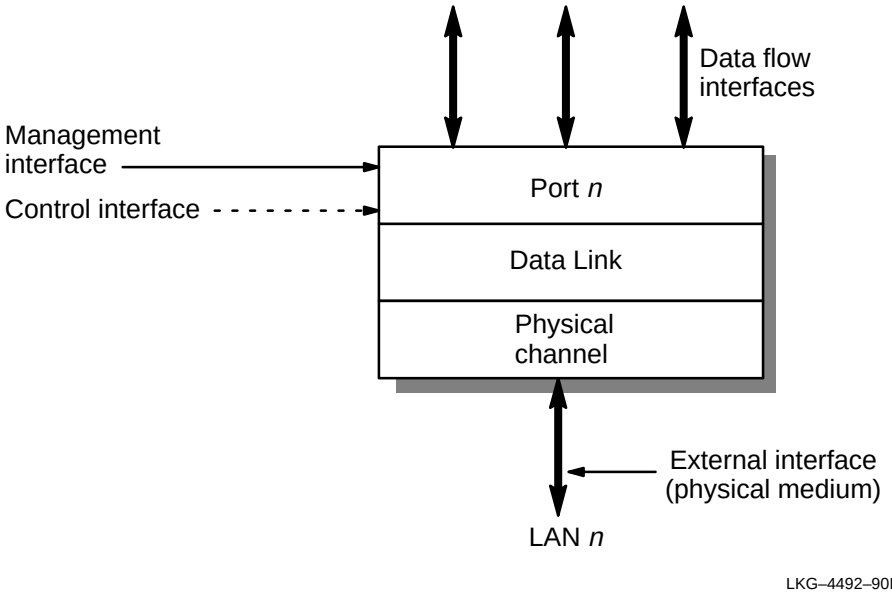
- **Data flow interfaces** — The data flow interfaces allow the bridge's functional modules to pass data frames.

The following sections provide detailed descriptions of the individual function modules that make up the bridge model.

2.1.2 Ports

The ports (see Figure 2–2) are the bridge's interface to its attached LANs. The ports consist of a data link section and a physical channel attachment section, and their associated interfaces.

Figure 2–2: Port Module Block Diagram



The data link section and physical channel attachment section correspond to the two lowest layers (Data Link and Physical) of the International Organization for Standardization model for Open Systems Interconnection, see Figure 1–4. These layers define the electrical and mechanical aspects of connecting to a physical medium. Together, they make it possible for devices to connect to a physical medium and to transmit and receive data over the channel.

Each port has a data interface to the forwarding and translating process module, management entity, and spanning tree entity. The ports pass frames (from packets received from the external interface) to these modules through these interfaces. Frames passed to the port from these modules are sent to the external interface.

2.1.2.1 Port States

The bridge ports contain the data link functions that provide the transmit and receive capabilities on a local network. The bridge ports are always in one of six states. These states are:

- **INIT** — The corresponding data link is initializing or testing itself. All data flow interfaces are off.
- **DISABLED** — The port is disabled by the management entity. The port is only available for receiving and transmitting management messages to and from the management entity.
  - The management entity interface is enabled for transmitting and receiving frames to and from the attached LAN.
  - The interface to the forwarding and translating process module is disabled.
  - The interface from the forwarding and translating process module is disabled.
  - The spanning tree entity interface is disabled for transmit and receive operations.
- **BROKEN** — The port is broken and cannot be relied on to provide valid frame transmit and receive operations. All data flow interfaces are off.
  - The management entity interface is disabled for transmitting and receiving frames to and from the attached LAN.
  - The interface to the forwarding and translating process module is disabled.
  - The interface from the forwarding and translating process module is disabled.
  - The spanning tree entity interface is disabled for transmit and receive operations.
- **PREFORWARDING** — This state is divided into two substates, **LISTENING** and **LEARNING.**

In the first half of the PREFORWARDING state, the port is in the LISTENING state. During this substate, it only receives frames destined to it.

In the second half of the PREFORWARDING state, the port is in the LEARNING state. During this substate, the forwarding and translating process module is placing new source addresses in the address database and the bridge is about to begin the frame relay service. The ports will all remain in the PREFORWARDING state for a time interval controlled by the spanning tree entity.

- The management entity interface is enabled for transmitting and receiving frames to and from the attached LAN.
- The interface to the forwarding and translating process module is enabled (LEARNING substate only).
- The interface from the forwarding and translating process module is disabled.
- The spanning tree entity interface is enabled for transmit and receive operations.

- **FORWARDING** — This is the bridge's normal operation state. Normal frame relay service will be provided on an extended LAN spanning tree.
  - The management entity interface is enabled for transmitting and receiving frames to and from the attached LAN.
  - The interface to the forwarding and translating process module is enabled.
  - The interface from the forwarding and translating process module is enabled.
  - The spanning tree entity interface is enabled for transmit and receive operations.
- **BACKUP** — The spanning tree entity has disabled the data flow interface to the forwarding and translating process module. This port is not active in the frame relay service, but will continue to monitor the LAN for management and spanning tree information.
  - The management entity interface is enabled for transmitting and receiving frames to and from the attached LAN.

– The interface to the forwarding and translating process module is disabled.

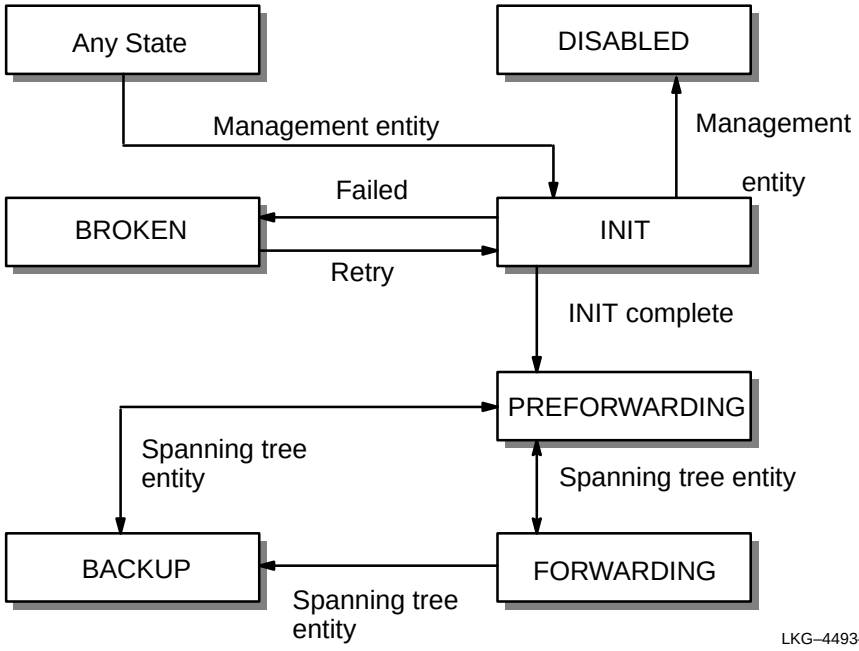
– The interface from the forwarding and translating process module is disabled.

– The spanning tree entity interface is enabled for transmit and receive operations.

2.1.2.2 Port State Transitions

The port can change from any other state to the INIT state if it determines that it would fail initialization. Figure 2–3 shows the port state transitions.

Figure 2–3: Port State Transitions



The management entity periodically attempts to restart a port that is in the BROKEN state.

2.1.2.3 Frame Aging

When a port dispatches a frame to the attached LAN, the frame is held resident in the port transmit queue until the data link can send the frame. This queue can become large when the port encounters congestion on the outbound LAN, thus causing frame transmit delays.

The frame aging (or *packet aging*) process ensures that the extended LAN does not contain data frames that were held resident in the transmit queue longer than the time limits required for certain Transport level protocol timers. The port deletes frames that are held resident too long in the bridge.

The frame age time is measured from the reception of the last byte of the CRC from the inbound LAN to the start of transmission of the destination address on the outbound LAN. Frames are held within the bridge no longer than 2 seconds.

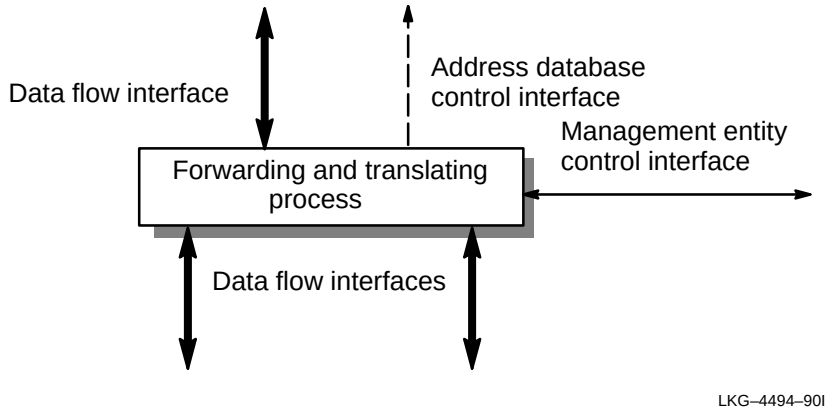
2.1.3 Forwarding and Translating Process Module

The forwarding and translating process module (see Figure 2–4) provides the frame forwarding function between bridge ports. This function, depending on the type of bridge, can include some or all of the following features:

- Frame Forwarding
- Translation
- Fragmentation
- Learning

After the bridge is initialized, the forwarding and translating process is active and has a number of counters and statistics that the management module can obtain through the management interface.

Figure 2–4: Forwarding and Translating Process Module Block Diagram



2.1.3.1 Frame Forwarding

The forwarding and translating process module uses its interfaces to the forwarding database module to determine whether a frame should be forwarded and which port should receive the frame. The bridge forwards frames to an outbound port if all of the following conditions are true:

- The frame was not received from that port.
- The source address filter is *not* set to discard the specific frame.
- The protocol type filter is *not* set to discard the specific frame.
- And any *one* of the following three conditions is also true:
  - The destination address of the frame is known to reside on that port.
  - The forwarding and translating process module does not know which port the destination address resides on (*flooding*).
  - The destination address is a multicast or a broadcast address (that has not been set to discard by bridge management).

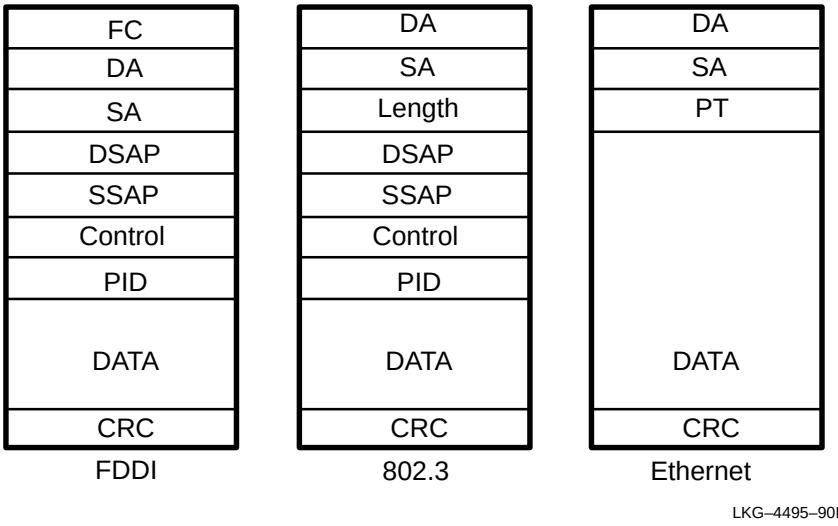
Digital's DECbridge 600 series multiport bridges have a special filtering feature that enables network management to specify forwarding/filtering criteria on a per-port basis. Chapter 6 contains examples of this feature.

Once the bridge decides to forward a frame, the frame is *translated* if the outbound port is connected to a dissimilar type LAN (refer to Section 2.1.3.2).

2.1.3.2 Translation

The DECbridge 500/600 series performs frame translation (rather than encapsulation) when forwarding a frame between the dissimilar LANs it interconnects. Frame translation means that frames that are to be forwarded are first converted into the native frame format of the destination LAN. Thus, translation type bridges allow stations on the FDDI LAN to communicate transparently with stations on the 802.3/Ethernet LAN. Note that the frame formats for FDDI, 802.3, and Ethernet are all different (see Figure 2–5).

Figure 2–5: Frame Formats



Translation from the IEEE 802.3 frame format to the FDDI frame format is fairly straightforward, since both FDDI and 802.3 frames should have an 802.2 Logical Link Control (LLC) header. However, when translating a frame from the Ethernet format to the FDDI format, an 802.2 LLC header must be be generated that conforms to Internet standard RFC 1042.

A standard 802.2 LLC header is created for Ethernet format frames to be sent on FDDI as follows:

- Both the Destination Service Access Point (DSAP) and Source Service Access Point (SSAP) values are set to Sub-Network Access Protocol (SNAP) AA hex.
- The Control field is set to U_I (Unnumbered Information), which has a value of 03.
- The Organizationally Unique Identifier (OUI), the first 3 bytes of the Protocol Identifier (PID), is set to 00-00-00 (the Ethernet OUI).
- The remaining 2 bytes in the PID are set to the Ethernet Protocol Type (PT).

When translating a frame from the FDDI format to the Ethernet format, the process is reversed to generate an Ethernet format frame from an FDDI frame.

Digital has added a special feature to translating bridges to help ensure that 802.3/Ethernet packets transferred between pairs of bridges will be retranslated into their original format. A table in each bridge contains the PIDs of special protocols that could otherwise be retranslated into the wrong format. The AppleTalk PID is entered into the table by default.

2.1.3.3 Fragmentation

The DECbridge 500/600 series is capable of performing IP fragmentation on IP packets received on the FDDI that are to be forwarded on the Ethernet, but are larger than the maximum packet size that Ethernet supports.

The maximum FDDI packet size, including the CRC, is 4495 bytes. The maximum Ethernet packet size is 1518 bytes. RFC 791, the standard that describes the Internet Protocol, specifies the rules for fragmentation when there is a mismatch in maximum data link size between the source and destination data link.

The DECbridge 500/600 series breaks up the received packet into legal Ethernet frames in accordance with this specification. An IP header is generated for each fragment generated, and re-assembly of these fragments is done by the destination node.

Note that IP fragments are generated only when the IP fragmentation switch is enabled (by the management software), the Don't Fragment (DF) bit in the IP header is clear, and the IP header is error-free. The IP fragmentation switch can be used to disable fragmentation and is enabled by default. If the switch is enabled but fragmentation is not occurring, the reason for dropping the IP packets can be detected by examining the IP fragment-related counters. See the documentation on your network management software for more information.

2.1.3.4 Learning

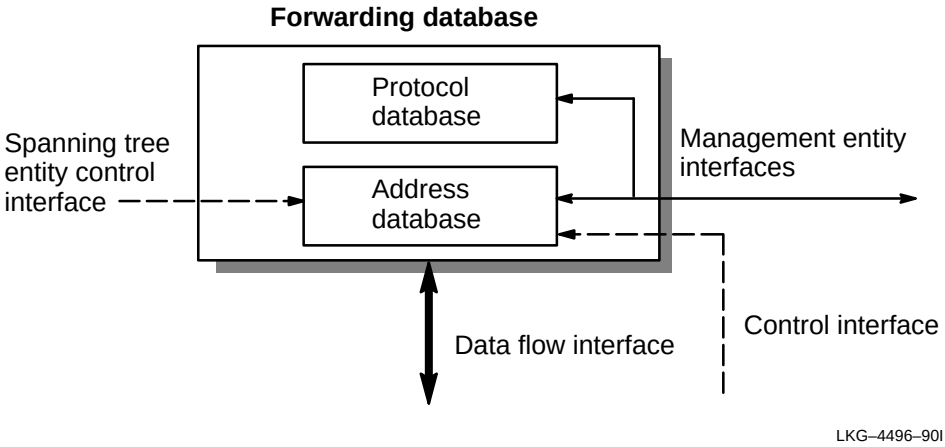
The forwarding and translating process module is responsible for adding unknown source addresses to the address database. The module uses the control interface to the address database for this function. When a frame is received from an attached port, both source and destination addresses are checked for their disposition in the address database. If the source address is not listed, it is added to the address database with its associated port address. While new source addresses can be added by the forwarding and translating process module, only the bridge management entity can set a source address permanently in the table with special status, such as, *not allowed on a given port*.

The address database deletes normally learned addresses from the database if the source addresses are not heard within a specified time. This timer value is provided to the address database by the spanning tree entity. Entries set by bridge management are not *timed out*.

2.1.4 Forwarding Database

The forwarding database (see Figure 2–6) has two distinct entities: the protocol database and the address database. The protocol database is optional and may not be present on all bridges, or it may only be partially implemented. The address database is required in all bridges.

Figure 2–6: Forwarding Database Block Diagram



2.1.4.1 Protocol Database

The protocol database is used by the forwarding and translating process module to help decide the disposition of frames received from the attached ports. The protocol database is controlled only by the bridge management entity, which is responsible for adding or deleting the protocol entries from the database.

2.1.4.2 Address Database

Upon initialization, the following addresses can be in the address database:

- The bridge's own physical address(es)
- Digital's "All Bridge Multicast Address" = 09-00-2B-01-00-00
- Digital's LAN Bridge 100 Spanning Tree Multicast Address = 09-00-2B-01-00-01
- The 802.1d Spanning Tree Multicast Address = 01-80-C2-00-00-00
- Addresses to be filtered as specified in the IEEE 802.1d specification
- Addresses specified by the management entity

The management interface to the address database allows the management entity to determine the number of addresses in the database, address age, address value, and port association. It can also set other available information, including setting an entry, to reside permanently in the address database.

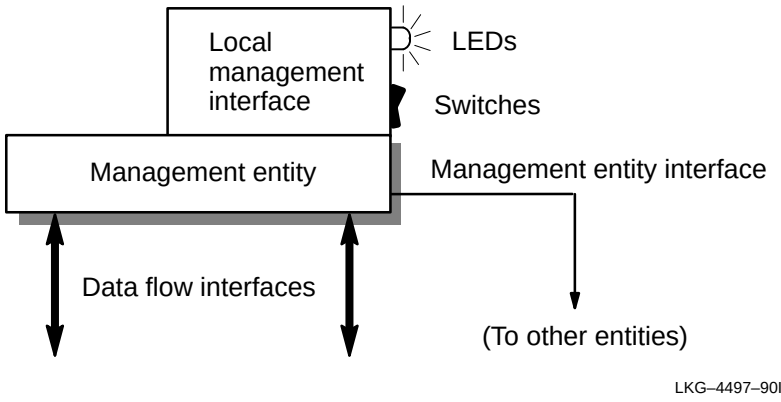
2.1.5 Management Entity

The management entity (see Figure 2–7) has management responsibility for the operation of the bridge and provides control and status input to each function module within the bridge.

It provides the following capabilities:

- Initializing (and self-test)
- Status and statistics gathering
- Setting state
- Setting information or characteristics
- Enabling or disabling a function
- Providing remote management capability
- Providing a local management interface

Figure 2–7: Management Entity Block Diagram



2.1.5.1 Protocol Database Access

The management entity has control access to the protocol database. The management entity is the only entity that can add or delete entries. The protocol database is always initialized with no entries set (all protocols forwarded).

2.1.5.2 Address Database Access

The management entity has control access to the address database. The management entity can add a number of *known* addresses to the address database for specific use.

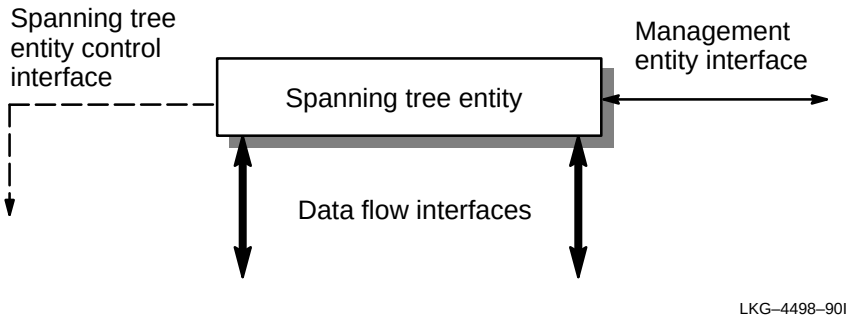
2.1.5.3 Spanning Tree Entity Access

The management entity has control and status access to the spanning tree entity. The management entity uses its interface to the spanning tree entity to determine the spanning tree characteristics, the root of the spanning tree, and other spanning tree entity information. This interface is also used to set other spanning tree parameters, such as priority and line cost.

2.1.6 Spanning Tree Entity

Extended LAN configurations cannot include any data link loops. If a loop were to occur, packets transmitted onto the extended LAN could circulate around the loop indefinitely. To prevent data link loops from occurring, the bridge has a special built-in spanning tree entity that can detect the looped configuration. The bridge uses the spanning tree entity to disable the port's interface to the forwarding and translating process module for any path involving a loop in an extended LAN topology. The spanning tree entity (see Figure 2–8) provides this function in a bridge.

Figure 2–8: Spanning Tree Entity Block Diagram



The spanning tree computation process provides the following features that enhance extended network capability:

- Loop detection
- Automatic backup (using redundant bridges)
- Determinism
- Low network overhead
- Self-maintenance
- Management

The spanning tree entity's control interface to the bridge ports is used to gain control over the state of any bridge port. Refer to the Port State diagram (see Figure 2–3) for these states.

The spanning tree entity's control interface into the address database is for setting the address age timer value. Typically, this value is 2 minutes but is shortened to 30 seconds when the spanning tree entity detects a reconfiguration of the spanning tree.

The management entity has a control input interface to the spanning tree entity. This control input interface is used by bridge management to gather state and status information and to control input, such as setting the root priority.

2.1.7 Bridge States

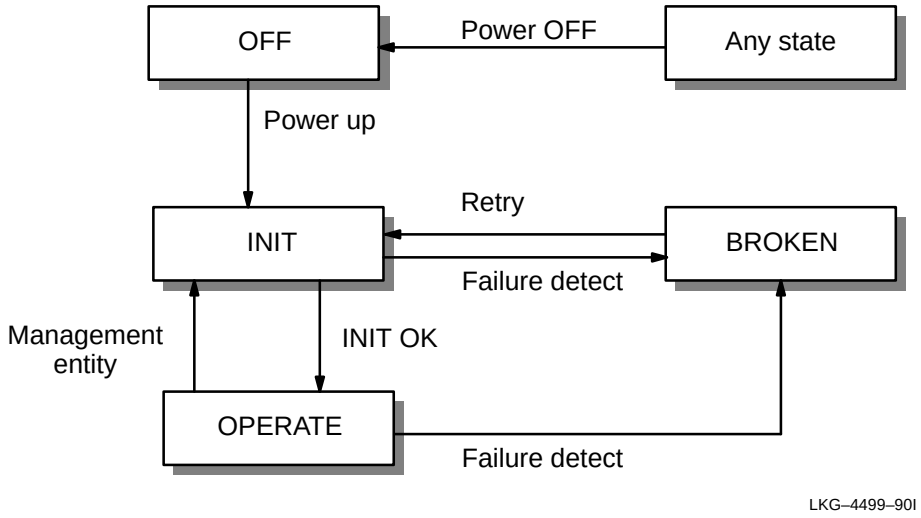
The bridge can be in one of four states during its operation (note that bridge states are distinct from the port states described in Section 2.1.2.1):

- **OFF** — Power is OFF. The bridge is not operating.
- **INIT** — The bridge is initializing all of the functional modules. This can include a self-test of each module.
- **OPERATE** — The bridge is performing all of its functions and at least one of its ports is operational.
- **BROKEN** — The bridge has detected a fatal error condition within itself.

The bridge can transition between states as shown in Figure 2–9.

When powered on, the bridge automatically enters the INIT state. It can go back to the OFF state only by being powered off. If the initialization is successful, the bridge enters the OPERATE state. If the initialization fails, the bridge reenters the INIT state or goes to the BROKEN state, depending on the nature of the failure. The bridge only stays in the BROKEN state for fatal errors; otherwise, it periodically retries initialization.

Figure 2–9: Bridge State Transitions



The management entity can reset the bridge, however, not all of these states may be available through the bridge management interface. For example, if the bridge is in the BROKEN state, the management entity could not remotely communicate this information.

The bridge can change states from OPERATE to INIT at any time if it determines that it would fail initialization and test.

The states shown for the bridge are effectively the combined states of the ports and the proper operation of all the bridge functions. For example, if all ports *were* in the BROKEN state, then the bridge (since it could not provide the relay service at all) would also be in the BROKEN state.

If any port *was not* in the BROKEN state *and could be properly initialized*, the bridge would go to the OPERATE state to allow the management entity to provide information on the current bridge status to network managers.

3

The Spanning Tree

This chapter explains how the spanning tree algorithm configures and maintains the spanning tree in an extended LAN. An implementation of the algorithm is contained in the spanning tree entity shown in the bridge model (see Figure 2–1).

Every extended local area network of any mesh complexity is logically configured into a network topology called a **spanning tree**. This is accomplished by a continuous, distributed process that is determined by the **spanning tree algorithm**. The spanning tree algorithm ensures that the configuration contains no loops (that there is only one path between any two nodes) and that all LANs are connected.

This chapter assumes that bridges have only two ports, although all concepts and procedures discussed in the chapter can be applied to multiport bridges.

3.1 The Spanning Tree Algorithm

This section discusses the two different implementations of the spanning tree algorithm used by Digital's bridges, the properties of the algorithm, and how the algorithm computes the spanning tree.

3.1.1 Implementations of the Spanning Tree Algorithm

Digital bridges use either of two implementations of the spanning tree algorithm: the implementation used by Digital's LAN Bridge 100 or the implementation described by the IEEE 802.1d MAC Bridge specification. Both implementations produce identical spanning tree logical configurations for any network configuration.

Digital's bridges (such as the LAN Bridge 150, LAN Bridge 200, and DECbridge 500/600 series), have an auto-select feature (refer to Section 3.3) that enables them to function in either LAN Bridge 100 spanning tree mode or 802.1 spanning tree mode. These bridges can be in the same extended LAN with either LAN Bridge 100 or 802.1 bridges.

3.1.2 Properties of the Spanning Tree Algorithm

The spanning tree algorithm has the following properties:

- **Loop detection** — If bridges are accidentally configured in a loop, the algorithm computes a loop-free portion of the topology.
- **Automatic backup (using redundant bridges)** — Bridges can be deliberately configured in a redundant path so that one of the bridges in the loop can serve as the backup for another. The process automatically configures a redundant bridge as a backup bridge. The backup bridge does not forward frames.
- **Determinism** — A fixed set of rules controls the process so that when variables change, the results are predictable.
- **Low network overhead** — The messages that control the spanning tree are usually transmitted at 1-second intervals (default), thus using a very small percentage of the available network bandwidth.
- **Management** — The algorithm allows tuning of parameters by management to control the topology.

3.2 The Spanning Tree Computation Process

The spanning tree computation process is continuous, setting up the spanning tree when bridges are initialized and maintaining the spanning tree afterward. Establishing the spanning tree involves these steps:

1. Bridges in the extended LAN elect a unique root bridge.
2. Bridges in the extended LAN elect a designated bridge for each LAN.
3. Redundant paths are removed from the logical spanning tree.

The spanning tree is self-maintaining, and performs these functions after it is established:

- Replaces a broken forwarding bridge with a backup bridge.
- Removes a redundant bridge when a loop is detected.
- Maintains address timers that control the aging of forwarding database address entries.

3.2.1 How Bridges Communicate with Other Bridges

The spanning tree algorithm is a distributed process in which all bridges in the extended LAN participate. Each bridge maintains information about itself and the spanning tree in databases: one database for the bridge spanning tree parameters (described in Section 3.4) and one database for each bridge port for port spanning tree parameters (described in Section 3.5). These parameters are used in computing the spanning tree and in providing results of the spanning tree computation.

Bridges communicate with each other with a minimum-size packet called a **Hello message**, referred to in the IEEE 802.1d specification as a Configuration Bridge Protocol Data Unit (BPDU). The Hello messages provide the following information:

- Best Root
- Cost to Root
- Designated ID
- Designated Port
- Root Age
- Forward Delay
- Listen Time
- Hello Time
- Hello Flags

Refer to Section 3.4 for a description of the bridge spanning tree parameters.

3.2.2 Determining the Root Bridge

Each spanning tree has a unique *root bridge*. The root bridge initiates Hello messages that propagate to other bridges in the extended LAN and dictates the values of certain spanning tree parameters for other bridges.

The root bridge for the spanning tree is the bridge with the lowest bridge ID. A bridge ID consists of the value of the bridge's Root Priority spanning tree parameter and its address. (Each port of a multiport bridge has a different address; the bridge address is one of the port data link addresses.)

The bridges in the extended LAN determine the root bridge by following these steps:

1. When a bridge is powered up, it sends out a Hello message on each port, claiming to be the root bridge of the extended LAN. The bridges adjacent to the sending bridge receive its message. Figure 3–1 illustrates the propagation of Hello messages among three bridges in a network segment at initialization.

   In this network segment, Hello messages sent by bridge B1 are received by B2; Hello messages sent by bridge B2 are received by B1 and B3; and Hello messages sent by bridge B3 are received by B2.

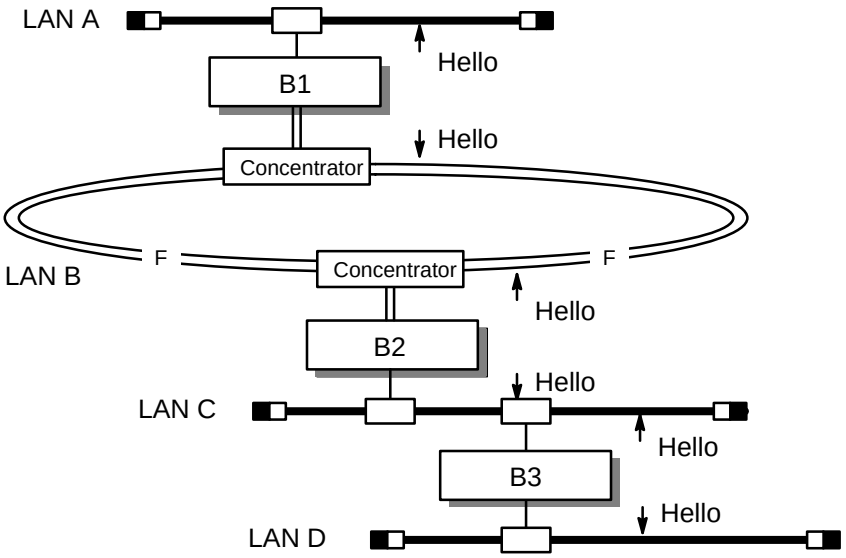
   This Hello message identifies the sending bridge as the root bridge by specifying its bridge ID as the Best Root parameter. The sending bridge stores this parameter in its spanning tree database.

2. When a bridge receives a Hello message, it compares the Best Root value in the message to the value of the Best Root parameter in its spanning tree database.

   a. If the Best Root in the incoming Hello message contains better root information (a lower bridge ID), the bridge determines that the bridge that sent the Hello message is the root bridge.

      The receiving bridge ceases to declare itself the root, stores the new root information that it received in its spanning tree database, and marks the port on which it received the better Hello message as the **inlink**. At the next Hello interval, this bridge will send a Hello message, on the other port, that identifies the new root bridge as the Best Root.

Figure 3–1: Propagation of Hello Messages at Initialization



B = Bridge

F = Fiber optic cable

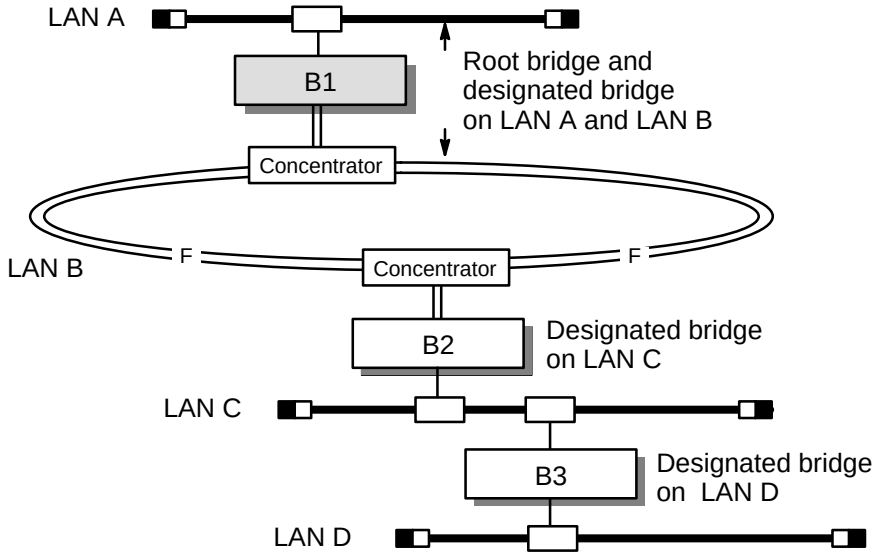
LKG–5393–90I

b. If the Best Root in the Hello message does not contain better root information, the bridge will continue to send its own Hello message at each Hello interval on both ports.

   The root bridge sends a Hello message at each Hello interval. Each bridge sends a Hello when it receives one on its inlink. At each subsequent Hello interval, each bridge sends a Hello message that declares the root bridge in the Best Root field.

3. Eventually, each bridge in the spanning tree knows the bridge ID of the root bridge. The root bridge initiates Hello messages, sending them on both its ports. Other bridges receive these messages on their inlinks and generate their own Hello messages, sending the messages to other bridges further from the root. Figure 3–2 shows the results of the process after stabilization.

Figure 3–2: Determination of Root Bridge



B = Bridge

F = Fiber optic cable

LKG–5394–90I

3.2.3 Determining Designated Bridges

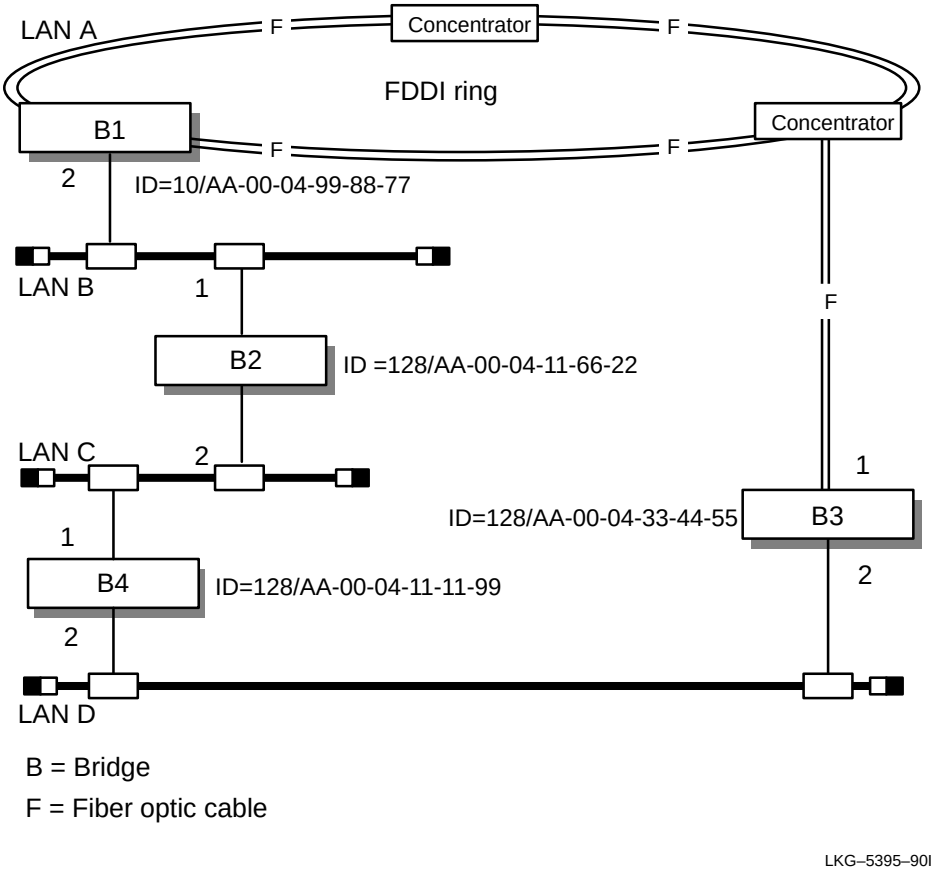
The **designated bridge** for a LAN is the bridge that logically connects the LAN to the next LAN closer to the root. This means that it provides forwarding service for that LAN. Each LAN has only one designated bridge.

Each port that connects a bridge to a LAN has a **line cost**, which is a value associated with the port. The **root path cost** for a bridge is the sum of the line costs for each inlink between the bridge and the root bridge. A designated bridge for a LAN is the bridge connected to the LAN that has the lowest root path cost to the root bridge.

If the root path cost for two bridges is the same, the designated bridge is the bridge with the lower bridge ID (tie-breaker). The root bridge is the designated bridge for both LANs it connects; all other bridges may be the designated bridge on any port but the bridge's inlink.

Figure 3–3 shows an extended LAN that consists of four bridges and four LANs. The figure labels bridge ports and marks each bridge with its bridge ID; the Root Priority precedes the bridge's address. All line costs are 10, their default value.

Figure 3–3: Determining the Root and Designated Bridges in a Looped Configuration



The following sequence describes how this extended LAN stabilizes.

1. At the first Hello interval, the following occurs:

   a. B1 receives Hello messages from B2 and B3 and compares addresses to determine the Best Root. B1 has the lowest bridge ID because its Root Priority is lowest, so it will continue to declare itself the root bridge and the designated bridge for LANs A and B in its next Hello message.

   b. B2 receives Hello messages from B1 and B4 and compares addresses, determining that B1 is the Best Root. Thus, port 1 is its inlink. B2 will continue to declare itself the designated bridge for LAN C and will declare B1 as the Best Root in its next Hello message. At the next Hello interval, B2 will send a Hello message on port 2 to B4.

   c. B3 receives Hello messages from B1 and B4 and compares addresses, determining that B1 is the Best Root. Thus, port 1 is its inlink. B3 continues to declare itself as designated bridge for LAN D and declares B1 as the Best Root in its next Hello message. At the next Hello interval, B3 sends a Hello message on port 2 to B4.

   d. B4 receives Hello messages from B2 and B3 and compares addresses, determining that B4 is the Best Root because it has the lowest bridge ID (priorities are the same). B4 continues to declare itself the designated bridge for LAN C and LAN D and declares itself as the Best Root in its next Hello message.

2. At the second Hello interval, the following occurs:

   a. B1 sends its Hello message to B2 and B3 (as before).

   b. B2 receives Hello messages from B1 and B4, indicating no change in information.

   c. B3 receives Hello messages from B1 and B4 (the latter declaring B4 as the Best Root). B3 compares addresses and determines that B1 is the Best Root. At the next interval, B3 continues to declare B1 as the Best Root and itself as the designated bridge for LAN D.

   d. B4 receives Hello messages from B2 and B3 on different ports, both declaring B1 as the root bridge, indicating that B4 is in a loop. B4 must determine whether it is the designated bridge for either LAN C or LAN D.

Both B3 and B4 attempt to become designated for LAN D. B3 becomes the designated bridge because its Cost to Root is lower (10) than B4's Cost to Root (20), even though it has a higher bridge ID. Thus, B4 becomes a backup bridge. B4's inlink is port 1, the port on the side of the bridge with the lower bridge ID (tie-breaker, because the cost to the root is the same on both sides). Port 1 moves to the FORWARDING state and port 2 moves to the BACKUP state.

3. At the third Hello interval, the following occurs:

   a. B1 sends its Hello message to B2 and B3 (as before).

   b. B2 receives B1's Hello message and detects no change.

   c. B3 receives B1's Hello message and detects no change.

   d. B4 receives Hello messages on both ports but detects no change. B4 remains a backup bridge.

      This Hello message pattern continues until a change is detected.

3.2.4 Port States

At initialization, the spanning tree entity sets each port to the PREFORWARDING state, because by default it is designated on each port (designated on the LAN to which the port is connected). During the forward delay, the port state of the inlink and of the port connecting a LAN to its designated bridge is PREFORWARDING. After the forward delay, the state changes to FORWARDING.

A port that is not the inlink and is not designated for any LAN is set to BACKUP. A bridge with a port in the BACKUP state does not forward frames on either port (two-port case). A bridge can forward frames only between ports in the FORWARDING state.

In the IEEE 802.1d specification, the BACKUP state is called *blocking*, the first half of the PREFORWARDING state is called *listening*, and the second half of the PREFORWARDING state is called *learning*.

Port states are described in more detail in Chapter 2.

3.2.5 One-Way Connectivity

The condition in which a bridge has a broken receiver or transmitter is called *one-way connectivity.* Traffic through a bridge with one-way connectivity moves in only one direction. The spanning tree computation process detects the problem in the following way:

- Broken Receiver — If a bridge receives no frames on a port for a specific period of time (defined by the No Frame Interval bridge spanning tree parameter), it runs a link test.

  If the receiver is broken, the port is brought down by setting its state to BROKEN. (This condition may cause the spanning tree to be recomputed.)

- A broken transmitter or a broken receiver on an adjacent bridge — Only the root bridge initiates Hello messages. Other bridges receive Hello messages on their inlink and send them on the port on which they are designated.

  If a designated bridge receives a Hello message on the port on which it is designated, and the message contains worse root information, that message is called a *bad Hello message.* A bad Hello could be sent by a bridge coming online after the extended LAN has stabilized (a normal situation that corrects itself). It could also indicate one-way connectivity in which the receiving bridge has a broken transmitter or the sending bridge has a broken receiver.

The spanning tree algorithm uses several bridge and port spanning tree parameters that count bad Hellos and the time intervals between them to distinguish between a normal and problematic condition:

- **Bad Hello Count** —The number of bad Hellos.
- **Bad Hello Limit** — The number of bad Hellos that, when reached, triggers a link test (which may indicate a defective transmitter).
- **Bad Hello Limit Exceeded Count** — The number of times the Bad Hello Limit has been reached since initialization.
- **Clear Time Count** —The number of successive Hello intervals during which no bad Hello was received.

- **Bad Hello Reset Interval** — The number of consecutive Hello intervals without a bad Hello. When the Clear Time Count reaches this value, the Bad Hello Count is reset. This allows for bad Hellos during initialization of the extended LAN.

- **Possible Loop Flag** — Indicates whether the Best Root named in the bad Hello message is the same as the Best Root known to the bridge receiving the bad Hello.

  If the Best Root values are the same, a loop would exist if the bridge sending the bad Hello and the bridge receiving it continue to forward.

These parameters work together in the following way:

1. When a bridge first receives a bad Hello message, it increments the Bad Hello Count parameter.

2. During each subsequent Hello interval, the bridge increments the Bad Hello Count if it receives a bad Hello or it increments the Clear Time Count if it does not receive a bad Hello.

   If the Best Root in the bad Hello is the same as the Best Root known by the bridge receiving the bad Hello, the receiving bridge sets the Possible Loop Flag; otherwise, the bridge resets the flag.

3. After each Hello interval, the spanning tree computation process checks these counters against the Bad Hello Limit and the Bad Hello Reset Interval.

   a. If the Clear Time Count reaches the Bad Hello Reset Interval, the bridge resets the Bad Hello Count and the Clear Time Count.

   b. If the Bad Hello Count reaches the Bad Hello Limit, the bridge increments the Bad Hello Limit Exceeded Count and performs a link test on the port receiving the bad Hello messages to determine whether its transmitter is working. The bridge resets the Bad Hello Count and Clear Time Count parameters.

      If the link test passes, the bridge sending the bad Hello messages probably has a broken receiver. This problem will be detected when that bridge receives no frames for the number of seconds specified by the No Frame Interval parameter (the test described previously).

If the Possible Loop Flag is set, a loop will exist if both bridges continue forwarding. To avoid a loop, the bridge receiving the bad Hellos stops forwarding frames (enters the PREFORWARDING state on the port receiving the bad Hellos).

If the bridge sending the bad Hello messages determines that it has a broken receiver, that bridge will stop forwarding and enter the BROKEN state on the faulty port. The bridge that received those messages will start forwarding again.

3.2.6 Topology Changes

Each bridge maintains a forwarding database that contains station addresses, the port on which the addresses are located, and the age of these entries (the number of seconds since the last time the bridge received a frame from this station). A bridge forwards or filters frames based on the information in its forwarding database.

There is only one active path between two stations in a loop-free extended LAN. If a bridge breaks, stations on one side of the bridge cannot communicate with stations on the other side until the extended LAN topology changes to circumvent the faulty path (assuming that a redundant path exists between these stations).

A *topology change* occurs when a backup bridge becomes a forwarding bridge. The path between stations on either side of the broken bridge may change because those stations can now be reached through the new forwarding bridge. Bridges must update their forwarding databases as quickly as possible so that stations are not isolated for long.

The spanning tree entity of each bridge maintains two parameters that affect the maintenance of their forwarding databases: Forwarding Database Normal Aging Time and Forwarding Database Short Aging Time. These parameters determine the maximum age of an entry in the forwarding database.

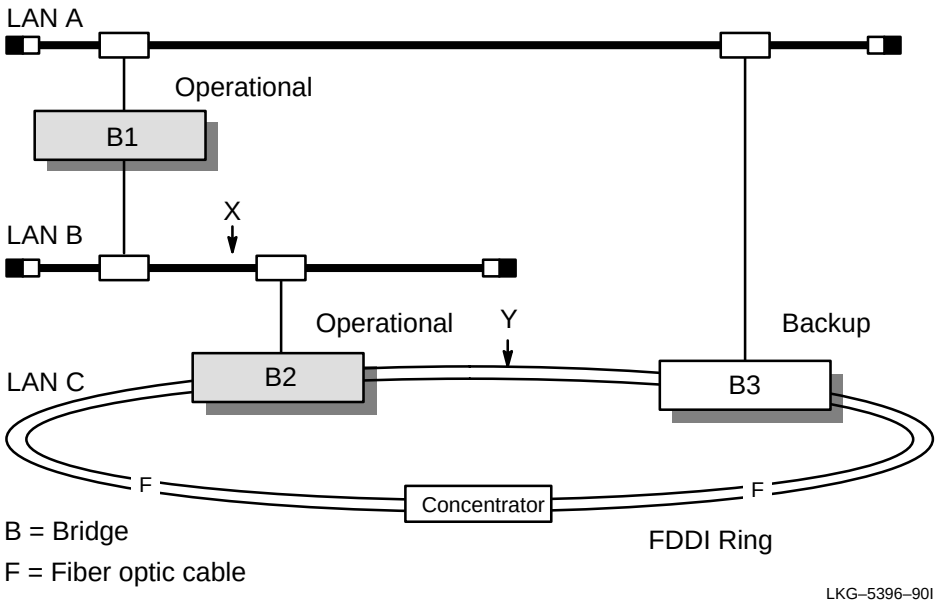
- Normal Aging Time — The amount of time a bridge retains an entry for a learned station address without receiving a frame sent by this station during normal network operation.
- Short Aging Time — The amount of time a bridge retains an entry for a learned station address without receiving a frame sent by this station following a topology change.

If the forwarding database holding time is too short, addresses are removed from the forwarding database too quickly and network performance suffers because too few frames are filtered. However, when a topology change has occurred, it is important to use a shorter address database holding time so that incorrect station information can be changed as quickly as possible.

When the new designated bridge starts forwarding, it sends a Topology Change Notification (TCN) to the root bridge on its inlink. Each intermediate bridge sends the TCN to the next bridge closer to the root and sets an Acknowledgment Flag, sent to the bridge that sent the TCN in its next Hello message.In the IEEE 802.1d specification, the TCN message is called TCN BPDU.

When the root bridge receives the TCN, it sets the Topology Change Flag in its Hello message and begins sending out the modified Hello message. As each bridge receives the Hello message, it reads the flag and switches from the Normal Aging Time to the Short Aging Time. The Topology Change Flag is set for the amount of time specified in the Topology Change Timer for the root. Figure 3–4 illustrates this process and the text following the figure describes the sequential process.

Figure 3–4: An Extended LAN Adjusting After a Topology Change



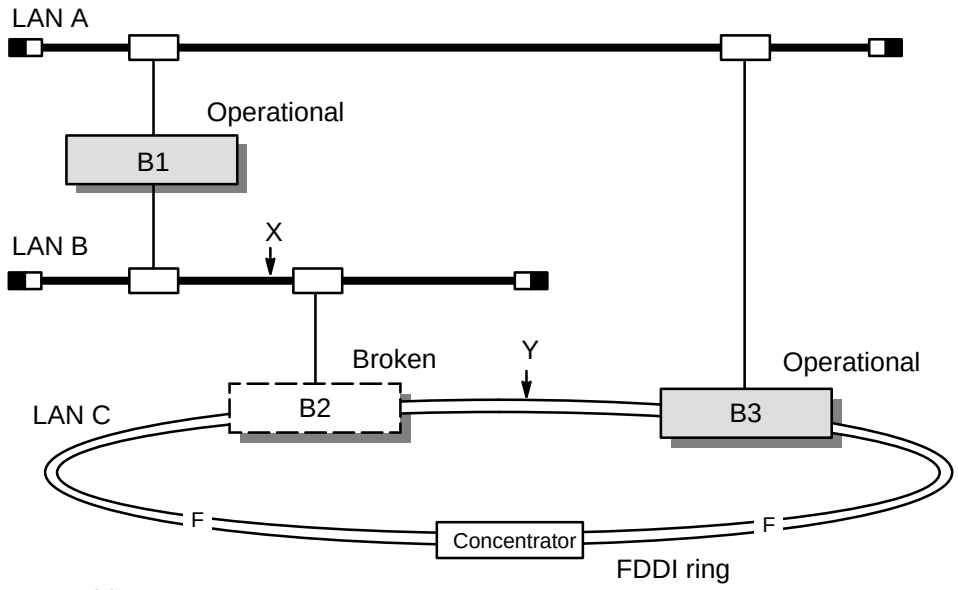
In Figure 3–4, bridges B1 and B2 are operational, and bridge B3 is a backup bridge. Station X can reach station Y through B2.

If B2 fails, station Y on LAN C cannot receive frames from station X on LAN B because the entry for Y in B1's forwarding database instructs B1 to filter frames received on LAN B addressed to Y.

When B3 begins forwarding, entries in B1's forwarding database for addresses on LAN C (including station Y) are aged out after the Short Aging Time. Bridge B1 learns that station Y can be reached on LAN A and adds or updates an entry for address Y in its forwarding database.

Station X can now communicate with Y much sooner than if the Normal Aging Time value had been used. Figure 3–5 shows the network segment after B3 replaces B2.

Figure 3–5: An Extended LAN Adjusted



3.2.7 Repeaters and Simple Bridges

A repeater is a device that forwards all frames it receives. A simple bridge is a bridge that may or may not be capable of filtering but does not participate in the spanning tree. Both repeaters and simple bridges will forward Hello messages like any other multicast message.

A network in which a simple bridge or repeater forms a loop causes the spanning tree bridge in the loop to stop forwarding. A loop consisting of only two simple bridges is undetectable and difficult to correct.

3.3 LAN Bridge 100, IEEE 802.1, and Auto-Select Bridges

LAN Bridge 100 models use Digital's LAN Bridge 100 implementation of the spanning tree algorithm. IEEE 802.1 bridges use the 802.1 spanning tree implementation, defined by the IEEE 802.1d, MAC Bridge specification. For a given physical topology, both implementations produce the same logical topology because the algorithm is the same in both.

Hello messages used by the two implementations are *not* compatible; LAN Bridge 100 models cannot interpret 802.1 Hello messages, and 802.1 bridges cannot interpret LAN Bridge 100 Hello messages. The LAN Bridge 150, LAN Bridge 200, and the DECbridge 500/600 series have an Auto-Select feature that enables them to function in either LAN Bridge 100 spanning tree mode or 802.1 spanning tree mode and interpret both types of Hello messages. Although these Auto-Select bridges can interpret both messages, the bridges can only operate in *one* spanning tree mode at a time.

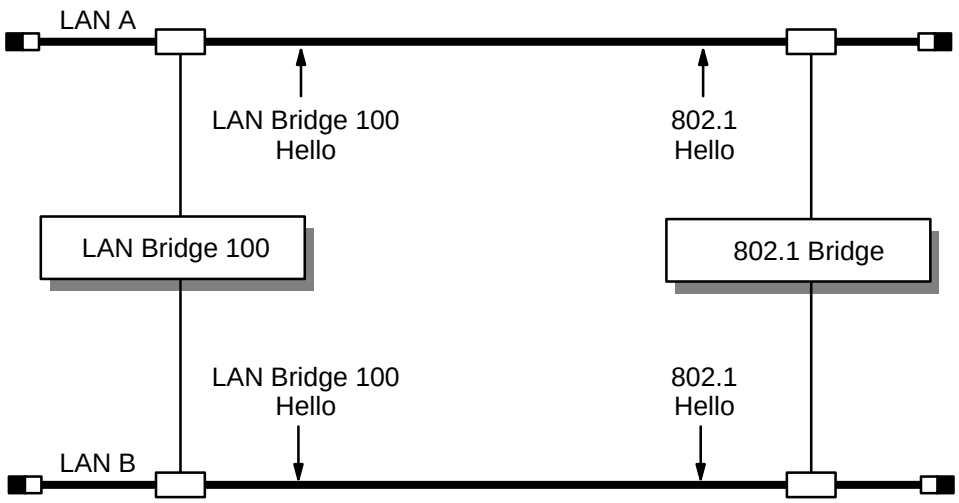
This section discusses the implications of having bridges that use both spanning tree implementations in the same extended LAN.

3.3.1 Combining LAN Bridge 100 and 802.1 Bridges in the Extended LAN

Because LAN Bridge 100 and 802.1 bridges cannot interpret each other's Hello messages, a network that contains both bridge types will produce two separate spanning trees. The LAN Bridge 100 Hello message is treated as a normal multicast message by an 802.1 bridge and forwarded. Similarly, 802.1 Hello messages are forwarded by LAN Bridge 100 models.

Figure 3–6 illustrates a simple topology consisting of two bridges, a LAN Bridge 100 and an 802.1 bridge, placed in parallel.

Figure 3–6: Potential Problem — A LAN Bridge 100 and an IEEE 802.1 Bridge in a Loop



NOTE: The above example applies to any mix of LAN types.

LKG–4505–90I

Both bridges send Hellos on LAN A and LAN B and try to start forwarding. In this topology, two problems can occur:

1. If the two bridges are powered up at the same time, have the same spanning tree timer values, and have a similar hardware base, neither would detect the other; they would start forwarding on both ports at the same time. If the two bridges send the next Hello message at the same time, they would both receive their own Hello message at the same time, indicating that they are in a loop. They would then enter the backup mode, expire themselves, and start forwarding together, repeating the cycle indefinitely.

   This condition is known as *thrashing* or oscillation. While these bridges thrash, there is no connectivity between the two LANs. In this situation, the topology does not stabilize if the timers continue to expire in each bridge at exactly the same interval.

2. If the bridges are not powered up at the same time, or if their timer values differ, the topology becomes stable, as only one bridge receives its Hello message and becomes a backup bridge. However, the resulting topology is not deterministic because it would depend on the order in which they were powered up.

   For example, if the 802.1 bridge is added to the network after the LAN Bridge 100 has started forwarding, the 802.1 bridge would become a backup bridge because the other bridge appears to be a repeater. It would receive its own Hello message and detect the loop.

This topology has the following results:

- Two spanning trees are formed. The topology cannot be managed effectively because not all bridges participate in the same spanning tree.
- The topology is not deterministic, which has these implications:
  - Network problems are difficult to diagnose. Since the topology is dependent on the order that the bridges are brought up, conditions are not always reproducible.
  - Network performance is unpredictable. The level of performance in various parts of the extended LAN will vary, depending on the current topology.
  - The topology cannot be tuned, since its behavior is not predictable.

3.3.2 Spanning Tree Auto-Select Bridges

Bridges having the Auto-Select feature (such as the LAN Bridge 150, LAN Bridge 200, and the DECbridge 500/600 series) are able to interpret both LAN Bridge 100 and 802.1 Hello messages. Thus, Auto-Select bridges can be combined with either LAN Bridge 100 or 802.1 bridges in a network.

An Auto-Select bridge can run either spanning tree implementation but can only use one at a given time. It may switch to the other mode if required, but it cannot be in both spanning trees at the same time.

By default, an Auto-Select bridge powers up as an 802.1 bridge in 802.1 spanning tree mode. However, as soon as it receives a LAN Bridge 100 Hello message, it switches to LAN Bridge 100 spanning tree mode. In LAN Bridge 100 spanning tree mode, an Auto-Select bridge filters 802.1 Hello messages.

The LAN Bridge 100 Spanning Tree Compatibility switch can force an Auto-Select bridge to operate in 802.1 spanning tree mode only. This switch should initially be set to True (Auto-Select mode) if any LAN Bridge 100 (or a bridge that configures with the LAN Bridge 100 spanning tree implementation) is present in the extended LAN. After migration of all bridges to the 802.1d standard is complete, the switch can be set to False (IEEE 802.1 spanning tree mode only).

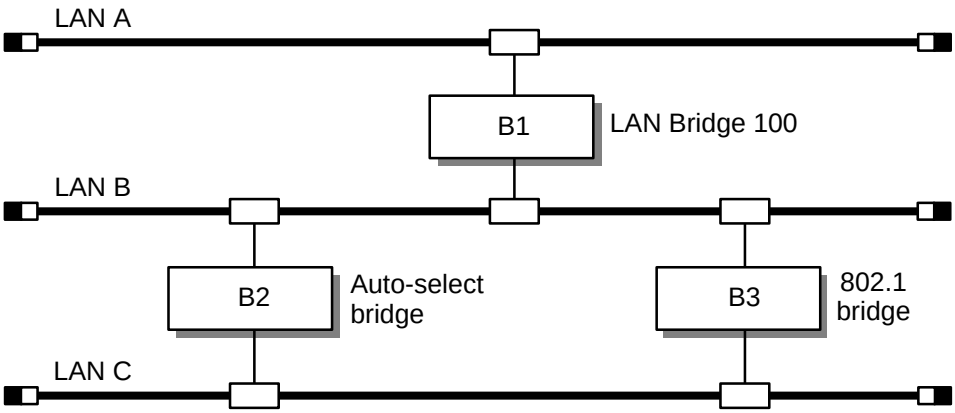
If the root bridge in a LAN Bridge 100 spanning tree is an Auto-Select bridge, it continuously determines whether it needs to remain in the LAN Bridge 100 spanning tree mode or can switch to the 802.1 spanning tree mode. The presence of any LAN Bridge 100 in the extended LAN keeps the Auto-Select bridge in the LAN Bridge 100 spanning tree mode.

The Auto-Select root periodically polls the LAN Bridge 100 to see if it is still present in the extended LAN. If the root gets no response to a poll, it assumes that the known LAN Bridge 100 has been removed from the network. It then sends out a multicast poll message to all bridges to determine whether another LAN Bridge 100 is present in the network. If any LAN Bridge 100 responds to the poll, the root remains in LAN Bridge 100 spanning tree mode and polls this LAN Bridge 100 periodically. If the root gets no response to the multicast poll, it switches to 802.1 spanning tree mode.

When the root changes over to 802.1 spanning tree mode, other Auto-Select bridges no longer receive the LAN Bridge 100 Hello messages and eventually switch to 802.1 spanning tree mode because the bridges conclude that the LAN Bridge 100 root has expired.

Figure 3–7 illustrates a network segment with LAN Bridge 100, Auto-Select, and 802.1 bridges.

Figure 3–7: LAN Bridge 100, IEEE 802.1, and Auto-Select Bridge in a Segment



NOTE: The above example applies to any mix of LAN types.

LKG–5398–90I

In this segment, B1 is the LAN Bridge 100, B2 is the Auto-Select bridge, and B3 is the 802.1 bridge.

Bridge B1 sends its Hello message onto LAN B.

Bridge B2 receives the LAN Bridge 100 Hello and switches to LAN Bridge 100 spanning tree mode, sending LAN Bridge 100 Hello messages.

Bridge B3 forwards the LAN Bridge 100 Hello message, since it cannot interpret it. Thus, B2, the Auto-Select bridge, receives its own Hello messages. Realizing that it is in a loop, B2 becomes a backup bridge.

As an Auto-Select bridge, B2 filters the 802.1 Hellos from B3, preventing B3 from receiving its own Hello messages. This happens deterministically and the two bridges never thrash.

It is not advisable to have LAN Bridge 100, 802.1, and Auto-Select bridges in the same extended LAN indefinitely. As shown in Figure 3–7, if the positions of the LAN Bridge 100 and Auto-Select bridges were exchanged, a LAN Bridge 100 would be in a loop with an 802.1 bridge, leading to indeterminism and, possibly, thrashing. This example topology may be seen during migration to the IEEE 802.1d standard.

3.4 Bridge Spanning Tree Parameters

Bridge spanning tree parameters describe and determine the spanning tree. Several bridge spanning tree parameters can be set with bridge management; others cannot be set but are determined by the spanning tree computation process and can be displayed with bridge management. This section discusses all of the bridge spanning tree parameters.

3.4.1 Actual Forward Delay

The Actual Forward Delay parameter indicates the Forward Delay currently in use by the root bridge. The Forward Delay for a bridge may be set with bridge management, but once the spanning tree computation process is complete, the bridge uses the root bridge's Forward Delay. If the bridge becomes the root bridge, its Forward Delay value becomes the Actual Forward Delay for all bridges in the network.

During the first half of the Forward Delay, bridges send and listen to Hello messages, participating in the spanning tree computation process. During the second half, bridges examine frames received on both ports, adding station address entries in their forwarding databases (this is the *learning process*).

3.4.2 Actual Hello Interval

The Actual Hello Interval parameter indicates the Hello interval currently in use by the root bridge. The Hello Interval parameter for a bridge may be set with bridge management, but once the spanning tree computation process is complete, each bridge uses the root bridge's Hello Interval parameter value. If the bridge becomes the root bridge, its Hello Interval parameter value becomes the Actual Hello Interval for all bridges in the network.

3.4.3 Actual Listen Time

The Actual Listen Time parameter indicates the Listen Time currently in use by the root bridge.The Listen Time for a bridge may be set with bridge management but, once the spanning tree computation process is complete, the bridge uses the root bridge's Listen Time. If the bridge becomes the root bridge, its Listen Time value becomes the Actual Listen Time for all bridges in the network.

3.4.4 Bad Hello Limit

The Bad Hello Limit parameter specifies the number of successive Hello intervals during which a bridge may receive bad Hello messages before the bridge performs a link test on the port. A bad Hello message may indicate a port problem. The Bad Hello Limit parameter works with other spanning tree parameters. Refer to Section 3.2.5 for more information.

3.4.5 Bad Hello Reset Interval

The Bad Hello Reset Interval parameter specifies how many Hello intervals without bad Hello messages a bridge will wait before it resets the Bad Hello Count for a port.

This parameter indicates how long a bridge will hold the current value of the Bad Hello Count, even though the bridge is not receiving bad Hello messages. The bridge automatically restarts this timer each time it receives another bad Hello message. The timer is expressed in Hello intervals. Refer to Section 3.2.5 for more information.

3.4.6 Best Root

The Best Root parameter indicates the bridge ID that this bridge believes is the root bridge. In the 802.1d specification, the name for this parameter is Designated Root. Refer to Section 3.2.2 for more information.

3.4.7 Best Root Age

The Best Root Age parameter indicates the age, in seconds, of the most recent Hello message from the Best Root. When the value of the Best Root Age exceeds the value of the Listen Time parameter, the bridge assumes the root has expired and sends out Hello messages on both its ports, declaring itself to be the root bridge and the designated bridge on its LANs. In the 802.1d specification, this parameter is called the Message Age.

3.4.8 Forwarding Database Normal Aging Time

The Forwarding Database Normal Aging Time parameter specifies how long a bridge retains learned station address entries in its forwarding database during normal network operation. If an address does not appear in the source field of a frame for a period of time defined by this parameter, its entry in the forwarding database is marked inactive and may be removed. Refer to Section 3.2.6 for more information.

3.4.9 Forwarding Database Short Aging Time

The Forwarding Database Short Aging Time parameter specifies how long a bridge retains learned station address entries in its forwarding database following a topology change. Refer to Section 3.2.6 for more information.

3.4.10 Forward Delay

The Forward Delay parameter specifies the period of time that a bridge's ports stay in the PREFORWARDING state before entering the FORWARDING state. This value is significant only for the root bridge, since it administers the Actual Forward Delay for all bridges in the spanning tree.

In the IEEE 802.1d specification, the Forward Delay is equivalent to the first half of the Forward Delay used in the Digital LAN Bridge 100 implementation, the time used by the bridge to listen to Hello messages.

3.4.11 Hello Interval

The Hello Interval parameter specifies the value of an interval timer that controls how often a bridge sends a Hello message. This value is significant only for the root bridge, since it administers the Actual Hello Interval for all bridges in the spanning tree.

3.4.12 Inlink

The Inlink parameter indicates the port number of this bridge's port on the path to the root bridge. This value is either 1 or 2 for a two-port bridge. In the IEEE 802.1d specification, this parameter is called the Root Port. Refer to Section 3.2.3 for more information.

3.4.13 LAN Bridge 100 Bridge Being Polled

The LAN Bridge 100 Bridge Being Polled parameter indicates the bridge ID of the LAN Bridge 100 that the Auto-Select root bridge is polling.

This parameter applies only to an Auto-Select root bridge operating in LAN Bridge 100 spanning tree mode. Refer to Section 3.3.2 for more information.

3.4.14 LAN Bridge 100 Poll Time

The LAN Bridge 100 Poll Time parameter specifies the number of seconds an Auto-Select root bridge waits between making LAN Bridge 100 poll attempts. The root polls the network to determine whether any LAN Bridge 100 models are present. If a LAN Bridge 100 responds within the time set by the LAN Bridge 100 Response Timeout parameter, the Auto-Select root stays in LAN Bridge 100 spanning tree mode and the bridge address is stored in the LAN Bridge 100 Bridge Being Polled parameter for the next poll.

This parameter applies only to an Auto-Select root bridge operating in LAN Bridge 100 spanning tree mode. Refer to Section 3.3.2 for more information.

3.4.15 LAN Bridge 100 Response Timeout

The LAN Bridge 100 Response Timeout parameter specifies the number of seconds an Auto-Select root bridge will wait for a response to a LAN Bridge 100 poll.

This parameter applies only to an Auto-Select root bridge in LAN Bridge 100 spanning tree mode. Refer to Section 3.3.2 for more information.

3.4.16 LAN Bridge 100 Spanning Tree Compatibility Switch

The LAN Bridge 100 Spanning Tree Compatibility parameter specifies whether the LAN Bridge 150, LAN Bridge 200, or DECbridge 500/600 series bridge is functioning as an Auto-Select bridge or as an 802.1 spanning tree bridge. Refer to Section 3.3.2 for more information.

3.4.17 Listen Time

The Listen Time parameter specifies the age of a Hello message (in seconds), after which the bridge considers the message to be *stale*. This value is significant only for the root bridge, since it administers Actual Listen Time for all bridges in the spanning tree. In the IEEE 802.1d specification, the name for this parameter is Max Age.

3.4.18 My Cost

The My Cost parameter indicates the bridge's current path cost to the root bridge. Refer to Section 3.2.3 for more information.

3.4.19 No Frame Interval

The No Frame Interval parameter specifies the number of seconds that a bridge waits without receiving a frame on a port before the bridge suspects a problem and runs a link test on the port. Refer to Section 3.2.5 for more information.

3.4.20 Root Priority

The Root Priority parameter is the most significant byte of the bridge ID. It can be used to establish the root bridge or designated bridge.

During the spanning tree computation process, Hello messages from all bridges in the network are compared so that the root bridge and designated bridges can be determined.The bridge with the lowest bridge ID becomes the root bridge, with Root Priority values compared first and hardware addresses second. Refer to Section 3.2.2 for more information.

3.4.21 Spanning Tree Mode

The Spanning Tree Mode parameter indicates whether the LAN Bridge 150, LAN Bridge 200, or DECbridge 500/600 series is operating in LAN Bridge 100 spanning tree mode or 802.1 spanning tree mode. Refer to Section 3.3.2 for more information.

3.4.22 Spanning Tree Mode Changes

The Spanning Tree Mode Changes parameter indicates the number of times that the mode of the spanning tree changed from 802.1 spanning tree mode to LAN Bridge 100 spanning tree mode. Refer to Section 3.3.2 for more information .

3.4.23 Tell Parent Flag

The Tell Parent Flag parameter indicates that the bridge needs to send a Topology Change Notification on its inlink to its parent bridge, the next closest bridge in the path to the root. Refer to Section 3.2.6 for more information.

3.4.24 Topology Change Flag

The Topology Change Flag parameter indicates that the root bridge has been notified of a topology change in the network and that bridges are to use the Forwarding Database Short Aging Time. Refer to Section 3.2.6 for more information.

3.4.25 Topology Change Timer

The Topology Change Timer indicates the number of seconds that the root bridge sends Hello messages with the Topology Change Flag set.

In LAN Bridge 100 spanning tree mode, the Topology Change Timer is the sum of the Forward Delay parameter value and the Short Aging Time parameter value. For 802.1 bridges, the duration is one-half the Forward Delay parameter value plus the Listen Time parameter value. Refer to Section 3.2.6 for more information.

3.5 Port Spanning Tree Parameters

Port spanning tree parameters consist of one settable parameter and several nonsettable parameters. Port spanning tree parameters describe the spanning tree from each port's perspective. The Line Cost parameter can be modified with bridge management; all others are nonsettable but can be displayed with bridge management. A database of port spanning tree parameters is associated with each port.

3.5.1 Acknowledgment Flag

The Acknowledgment Flag indicates that the port has received the Topology Change Notification from a bridge lower down in the spanning tree (further from the root bridge). The acknowledgment is sent in the next Hello message. In the IEEE 802.1d specification, this parameter is called the Topology Change Detected flag. Refer to Section 3.2.6 for more information.

3.5.2 Bad Hello Count

The Bad Hello Count parameter indicates the number of consecutive Hello intervals during which the bridge received a bad Hello message on a port. When the value of the Bad Hello Count reaches the Bad Hello Limit set for the bridge, the bridge resets this counter, increases the Bad Hello Limit Exceeded Count by one, and performs a link test on the port.

Note that if the Clear Time Count parameter value reaches the value of the Bad Hello Reset Interval bridge parameter *before* the Bad Hello Count reaches the Bad Hello Limit, the port resets the Bad Hello Count. Refer to Section 3.2.5 for more information.

3.5.3 Bad Hello Limit Exceeded Count

The Bad Hello Limit Exceeded Count indicates the number of times that this bridge's Bad Hello Limit has been exceeded since its initialization. Refer to Section 3.2.5 for more information.

3.5.4 Clear Time Count

The Clear Time Count parameter indicates the number of consecutive Hello intervals during which the bridge has received no bad Hello messages on this port. When the Clear Time Count reaches the Bad Hello Reset Interval bridge parameter, the bridge resets the Clear Time Count and the Bad Hello Count. Refer to Section 3.2.5 for more information.

3.5.5 Designated Bridge ID

The Designated bridge ID parameter indicates the bridge ID of the designated bridge on this LAN (the LAN connected to this port).

3.5.6 Designated Bridge Link Number

The Designated Bridge Link Number parameter indicates the port number of the designated bridge on this LAN (the LAN connected to this port).

3.5.7 Designated Root Age

The Designated Root Age parameter indicates the age, in seconds, of the last Hello message sent by the designated root.

3.5.8 Designated Root ID

The Designated Root ID parameter indicates the bridge ID of the bridge considered to be the root bridge on this port.

3.5.9 Forward Delay Timer

The Forward Delay Timer parameter indicates the time remaining before the port will leave the PREFORWARDING state and enter the FORWARDING state.

3.5.10 Line Cost

The Line Cost parameter specifies the cost value for the port, which is used to determine the path cost to the root bridge. The Line Cost parameter may be set by network management.

3.5.11 Port Address

The Port Address parameter indicates the hardware address of the port, which may differ from the bridge address.

3.5.12 Possible Loop Flag

The Possible Loop Flag parameter indicates whether the bridge has detected a loop condition in a situation where the Bad Hello Count is not zero. Refer to Section 3.2.5 for more information.

3.5.13 Root Path Cost

The Root Path Cost parameter indicates the cost to the root bridge for the designated bridge on this LAN.

4

Extended LAN and Bridge Management

Effective extended local area network (LAN) and bridge management means ensuring that the extended LAN conforms to the needs of network users and performs with optimum efficiency. The network manager can manage bridges, lines, and traffic flow. Management involves these activities:

- Configuring
- Controlling
- Monitoring
- Troubleshooting

This chapter provides an overview of some of the network manager's functions. Topics presented in this chapter are discussed in greater detail in Chapters 3, 5, and 6 of this manual, and in the network management software documentation.

Digital provides the following network management products that allow you to manage the bridges (and FDDI concentrators) in your extended network:

- DECelms
- DECmcc Extended LAN Manager Software
- DECmcc Management Station for ULTRIX

NOTE

Throughout this manual, the generic term *network management software* will be used to represent these network management products. For specific information on each product, refer to the associated manuals.

4.1 Configuring an Extended LAN

The spanning tree algorithm, described in Chapter 3, produces a logical network configuration from a physical arrangement of LANs and bridges. The algorithm ensures that the extended LAN contains no loops and that all LANs are connected. The network manager can affect the logical configuration that the spanning tree computation process produces, thereby affecting network performance.

4.1.1 Specifying the Root Bridge of the Extended LAN

The root bridge in an extended LAN is the bridge that controls the network configuration by originating Hello messages. The placement of the root bridge can affect network performance. The spanning tree algorithm determines the root bridge by comparing the bridge IDs for all bridges in the extended LAN. The root bridge is the bridge with the lowest bridge ID (Root Priority and hardware address). The process used by the spanning tree algorithm to determine the root bridge is described in detail in Section 3.2.2.

Since the hardware addresses are set at the factory, you can specify the root bridge by using bridge management to assign the lowest Root Priority value to the bridge that you want to become the root bridge.

4.1.2 Specifying Designated Bridges for LANs

A designated bridge for a LAN is the bridge that connects the LAN to the path to the root bridge. The spanning tree computation process automatically determines the designated bridge for each LAN. To minimize traffic through an inefficient LAN or through a heavily used LAN, it is important to ensure that the spanning tree algorithm chooses a particular bridge to become the designated bridge for a LAN.

Each line connecting a bridge to a LAN has a **line cost** that helps to determine the designated bridge for the LAN. The line cost is not a monetary cost but is, instead, based on issues such as line bandwidth, anticipated line traffic, and so on. The sum of the line costs for all inlinks between the LAN and the root bridge is the **root path cost**. The bridge with the lowest root path cost becomes the designated bridge for the LAN. If more than one bridge has the same lowest root path cost, the bridge with the lowest bridge ID becomes the designated bridge. The process used by the spanning tree algorithm to elect designated bridges is described in detail in Section 3.2.3.

You can cause the spanning tree computation process to select a particular bridge as the designated bridge for a LAN by using network management to set the Line Cost spanning tree parameter for one or more ports. Assign a low Line Cost for a bridge line if you want the bridge to become the designated bridge on the LAN attached to the line. Refer to Chapter 6 for more information about specifying designated bridges for LANs.

4.1.3 Spanning Tree Mode Selection

The spanning tree computation process developed by Digital Equipment Corporation was first implemented in Digital's LAN Bridge 100 product. This spanning tree algorithm was offered to the IEEE and has now been adopted by the IEEE as part of the IEEE 802.1d, MAC Bridging Standard.

Although the two algorithms are identical and produce exactly the same spanning tree topologies, the 802.1d standard has a slightly different implementation of the algorithm. For example, the multicast address used for the bridge Hello message is different in the LAN Bridge 100 spanning tree implementation than it is in the 802 spanning tree implementation. As a result, bridges that operate in one of these two spanning tree modes cannot understand Hello messages from bridges that operate in the other spanning tree mode.

To help solve this problem, the LAN Bridge 150, LAN Bridge 200 and DECbridge 500/600 series dynamically select either the 802.1 spanning tree mode or the LAN Bridge 100 spanning tree mode, depending on what the network configuration requires.

An Auto-Select software switch controlled by the bridge management software lets you either enable this *auto-configure* (or *migration*) mode or lock the bridge in 802.1 spanning tree mode.

4.2 Controlling Bridges and Ports

A network manager can control bridges and ports, affecting network traffic and bridge operation. This section discusses some of the ways the network manager can control bridges and ports.

NOTE

For specific information on how your network management software performs these functions, refer to the documentation on your particular management system.

4.2.1 Restricting Access to the Extended LAN

Each bridge sets up and maintains a forwarding database, which determines whether a bridge forwards or filters (discards) a frame that it receives. The forwarding database consists of an address database and, for certain bridge models, a protocol database. You can restrict access to portions of the extended LAN by using bridge management to add entries to the forwarding databases of one or more bridges.

4.2.1.1 Managing Address Entries in the Forwarding Database

During the forwarding delay, each bridge listens to traffic on its LANs. By examining the source address of each frame it receives, a bridge can determine which port the station is heard on. Then, when a bridge is forwarding and receives a frame addressed to a particular station, it can determine which port the frame should be sent to. Each bridge's address database contains a list of station addresses and the bridge line that indicates the side of the bridge where that station is located. The entry also indicates whether frames with that station address are to be forwarded or filtered.

To restrict a station's access to portions of the extended LAN, the forwarding database entry should be set in the bridge to filter frames for the user's station. LAN Bridge 100, LAN Bridge 150, LAN Bridge 200, and DECbridge 500/600 series can filter frames for a station address appearing in the destination field of the frame. The LAN Bridge 200 and DECbridge 500/600 series can also filter frames if the address appears in the source field of the frame. You can add forwarding database entries with bridge management.

4.2.1.2 Selective Address Forwarding

The LAN Bridge 200 and DECbridge 500/600 series allow you to control which stations can communicate across the bridge. This *selective address forwarding* feature is controlled by a management set software switch (manual filter software switch), which allows you to indicate the specific source and destination addresses to be forwarded and instruct the bridge to filter (discard) all other addresses. You can selectively control source and destination addresses only with bridge management.

4.2.1.3 Managing Protocol Entries in the Protocol Database

The LAN Bridge 200 and DECbridge 500/600 series units maintain a protocol database of protocol entries. Each entry identifies a frame type, protocol identifier, and defines whether the bridge is to forward or filter frames that contain that protocol. These bridges allow you to control traffic of frames using particular protocols by adding entries to their protocol database. You can add protocol database entries only with bridge management.

4.2.2 Disabling and Enabling Bridge Ports

You can remove a bridge port from the spanning tree by disabling it. Disabling the bridge port forces the spanning tree algorithm to recompute the spanning tree. (Note that disabling or enabling a LAN Bridge 100 or a LAN Bridge 150 port also causes these bridge models to initialize.) You can disable and enable bridge ports with bridge management.

4.2.3 Initializing Bridges

Initializing a bridge resets the bridge, just as if it had been physically turned off and then on again. If you suspect a problem with a bridge, you can initialize it to force it to perform self-tests. Also, if you modify spanning tree parameters incorrectly, you can initialize bridges and reset the parameters to their factory default values. You can initialize a bridge with bridge management.

4.2.4 Setting the Device Password

You can assign and set a password for each LAN Bridge 150, LAN Bridge 200, DECbridge 500/600 series, and DECconcentrator 500 in your extended LAN. (The LAN Bridge 100 does not support device passwords). The password is stored in the device's nonvolatile memory (NVRAM). A device that has a password set checks that the correct password is supplied with any network management configuration or control command and rejects any command with an incorrect or absent password. You can set (or change) device passwords only with bridge management.

4.2.5 Upline Dump of Memory Image

You can configure your LAN Bridge 200 for upline dumping, so that a memory image is sent to a DECnet-VAX host if a fatal error causes the device to crash. Digital Services can then analyze the dump file to determine the cause of failure. The documentation on your network management software explains how to set up a DECnet-VAX host to receive upline dumps, and also describes how to enable or disable upline dumping on the device, and how to indicate which host is to receive the upline dump file.

4.2.6 Downline Loading of Executable Images

You can configure your LAN Bridge 100 or LAN Bridge 150 so that it will request a downline load of software (such as LTM Listener) upon initialization instead of loading the bridge code stored in its read only memory (ROM).

You can configure your DECbridge 500/600 series and DECconcentrator 500 to accept downline loaded upgrades to the device operational code. The downline load process overwrites the image stored in the device's nonvolatile memory. Use the utility provided with the upgrade software for downline load upgrades.

4.2.7 Controlling IP Fragmentation on a DECbridge 500/600 Series

You can control the fragmentation of Internet Protocol (IP) frames on a DECbridge 500/600 series. Through the network management software, you can instruct the bridge to break up large Internet Protocol frames received on its FDDI line into smaller frames that can be transmitted on its IEEE 802.3/Ethernet line. This fragmentation is necessary because the maximum size for a frame on an FDDI ring is 4500 bytes, but only 1518 bytes for a frame on an IEEE 802.3/Ethernet segment. The documentation on your network management software explains how to enable and disable IP fragmentation, and how to display the settings of the software switch that controls fragmentation.

4.2.8 Setting the Target Token Rotation Time for an FDDI Line

You can use bridge management to set the target value that the FDDI MAC entity of a DECbridge 500/600 series or DECconcentrator 500 will bid in the claim token process. The claim token process sets the target token rotation time (TTRT) used by all stations on the ring and determines the station that will originate the token. See the documentation on your network management software for more information on how to set and display the TTRT parameter settings.

4.2.9 Setting the Valid Transmission Timer for an FDDI Line

You can use bridge management to set the Valid Transmission Timer (TVX) value for the FDDI MAC entity of a DECbridge 500/600 series or DECconcentrator 500, and display the value that is currently in effect. Each FDDI station has a TVX that detects token loss on the ring, excessive noise, and other faults. The station resets its TVX to zero upon receipt of the ending delimiter of a valid frame or nonrestricted token. The timer expires when the time since the last valid transmission exceeds the TVX value set for the station. When the valid transmission timer expires, the station starts the claim token process, which initializes the ring and creates a new token.

4.2.10 Setting the Link Error Monitor Threshold for a Physical Port

You can use bridge management to set the link error monitor (LEM) threshold for a physical port (PHY entity) on a DECbridge 500/600 series or a DECconcentrator 500, and display the current LEM threshold settings. The LEM checks the bit error rate (BER) on the physical port during normal operation. When the bit error rate rises above the LEM threshold, the station disables the physical port, preventing it from disrupting the ring. The station then repeatedly runs the link confidence test (LCT) on the physical port until the physical port is validated. The bit error rate on a physical port can rise because of marginal port quality, port degradation, or simply because the connector is loose.

4.2.11 Setting the Collision Presence Test Characteristic

If a bridge port has a transceiver that uses the Collision Presence Test (CPT), the CPT characteristic (often referred to as *heartbeat*) for the line should be set. You can set the CPT characteristic with bridge management. This maintainability feature is only useful if the CPT function is enabled when the Medium Attachment Unit (MAU), such as a DELNI, provides the function. The bridge will continue to work correctly with either setting.

4.3 Monitoring Bridges and Ports

Locating and correcting bridge and LAN problems involves examining network activity by monitoring bridges and ports. Counters provide useful statistics for monitoring network activity. Also, bridges can serve as traffic monitors, providing statistics that can be used to find problems.

4.3.1 Examining Bridge and Port Counters

Bridges and ports maintain counters that track frame traffic, errors, and other events. These counters keep track of frames and bytes received by the bridge, and can classify them in various ways to provide useful information for troubleshooting an extended LAN.

4.3.2 Using a LAN Bridge 200 as a LAN Monitor

A LAN Bridge 200 can also serve as a LAN monitor (not to be confused with LTM listener described in Section 4.3.3), allowing you to observe the utilization, throughput, and other characteristics of the LANs to which the bridge is attached. You can monitor a LAN by using a LAN Bridge 200 with bridge management.

4.3.3 Designating a LAN Bridge 100 as an LTM Listener

The LAN Traffic Monitor (LTM) is separately priced Ethernet monitoring software that uses a LAN Bridge 100 or LAN Bridge 150 to gather network traffic statistics and periodically forward them to a VMS system for compilation and analysis.

A bridge that is loaded with LTM Ethernet monitoring software is referred to as an **LTM listener**. Bridges serving as LTM listeners are dedicated solely to LTM; they do not perform normal bridge functions and are controlled by LTM software. You can designate a LAN Bridge 100 or a LAN Bridge 150 to be an LTM listener with bridge management.

4.4 Bridge Management and the Bridge Model

This section discusses how different entities, shown in the bridge model (see Figure 2–1), are involved in bridge management.

4.4.1 The Spanning Tree Entity

The spanning tree entity works with other entities in the following ways:

- Management commands can modify spanning tree parameters and cause recomputation of the spanning tree. The management entity provides these values to the spanning tree entity.
- The spanning tree entity controls port states during the computation of the spanning tree.
- Spanning tree parameters can control the status of entries in the forwarding address database.

4.4.2 The Management Entity

The management entity works with other entities in the following ways:

- Management commands can control the state of ports.
- Management commands can add, delete, and modify entries in the forwarding database (either the address database or the protocol database).

- Management commands can modify spanning tree parameters maintained in the spanning tree entity.
- Management commands can affect the bridge by initializing it, or they can display bridge counters or characteristics.

4.4.3 The Forwarding Database

The forwarding database consists of the address database, which contains station and multicast address entries, and the protocol database, which contains protocol entries.

The forwarding database entity can work with other entities in the following ways:

- During the forward delay, the forwarding and translation process module examines frames that are received by the bridge ports and adds station address entries to the address database from the source addresses in the frames.
- While the bridge is forwarding, the forwarding and translation process module checks the address database and the protocol database (for the LAN Bridge 200 and DECbridge 500/600 series) to determine whether to forward or filter frames that are received by the bridge.
- Management commands, passed on by the management entity, can add, remove, or modify entries in either the address database or the protocol database.
- The spanning tree entity can affect the status of address database entries with the Forwarding Database Normal Aging Time and Short Aging Time spanning tree parameters.

5

Bridge and Extended LAN Performance

High performance for extended local area networks and bridges means minimal delay and maximal data integrity. The performance goal is to provide extended LAN end nodes the same (or better) datagram service than would be received on a single LAN. This chapter discusses factors that affect the performance of extended LANs and bridges.

5.1 Extended LAN Performance

An important performance goal of an extended LAN is that it performs like a LAN. Ideally, users on an extended LAN should think that they are part of a single LAN and not be aware that they are part of a collection of LANs with connecting bridges. Some factors that can affect the performance of extended LANs and bridges are listed here and are described in the subsections that follow:

- End-to-end delay
- Diameter of the extended LAN
- Frame lifetime
- Frame loss (caused by data errors)
- Undetected data corruption
- Low-performance controllers
- Frame loss caused by congestion

5.1.1 End-to-End Delay

The **end-to-end delay** is the maximum amount of time it can take for a packet to travel from one end of the extended LAN to the other. The end-to-end delay is affected by the extended LAN diameter, by bridge processing, and by LAN congestion.

The **extended LAN diameter** is the number of LANs on the path between the two most distant stations. The maximum diameter is limited by the allowable end-to-end delay between any two stations. For delay-sensitive protocols, such as Local Area Transport or Terminal Server applications, the average one-way delay across the extended LAN should not exceed 10 milliseconds for a 100-byte packet.

5.1.2 Frame Lifetime and the Seven-Bridge Rule

The frame lifetime in an extended LAN is the amount of time it takes for a frame to reach its destination. The lifetime of a frame in an extended LAN must not exceed 15 seconds. Since frames cannot exist in a bridge for more than 2 seconds, the maximum suggested extended LAN diameter is seven bridges (eight LANs).

Note that this maximum extended LAN diameter is an architectural limit which allows Transport level LAN services to have a guarantee on frame lifetime. Transport timer and frame sequence numbers and size can be properly set once the Data Link frame lifetime is fixed. Many protocols, however, provide poor or unusable service to applications at times approaching even 10 percent of the maximum frame lifetime.

5.1.3 Frame Loss Caused by Data Errors

Frames may be lost in the extended LAN because of data errors. The average frame loss rate due to detected data errors must not exceed one frame in 10,000.

As each frame is sent, it is assigned a cyclic redundancy check (CRC) value that is determined by the frame contents. The CRC value is a frame check sequence, used for verifying the frame contents. When a bridge receives the frame, it calculates the CRC and compares that value with the value in the frame. If the values match, the frame is assumed to be correct and processing continues. If the values do not match, the frame is discarded because of a data error. Generally, data errors occur due to noise while the frame is traversing a LAN.

5.1.4 Undetected Data Corruption

A rare, but not impossible, event can occur when the CRC indicates that the received frame contains correct information, even though several data errors may have occurred in the frame. This event is known as *undetected data corruption*.

As the LAN grows to an extended LAN, the transport protocols expect to receive the same level of performance for an undetected data error rate. Therefore, as the bridge stores and forwards frames, it must not increase the probability that a user will receive a frame with undetected data corruption. An excellent method for performing this is to *preserve* the CRC.

As the frame is received, the bridge port checks the CRC for errors and discards any errors it finds. However, the received CRC is also placed in the bridge memory along with the frame. If the frame is transferred to another port in the bridge, the bridge sends the original CRC, which was stored in the bridge's memory, along with the frame (see note).

NOTE

Translating type bridges (such as the DECbridge 500/600 series) alter the incoming frame to comply with the format required by the outbound port's data link. Therefore, the received CRC cannot be preserved and transmitted to the outbound port. The forwarding and translation process module verifies the frame contents using the received CRC while *recalculating* a new CRC to reflect the added or altered information. This new CRC is sent along with the translated frame to the outbound port.

This method ensures that *any* errors that may have occurred to the frame while it was still in the bridge will be protected by the original (or recalculated) CRC value. This ensures the maximum data protection possible for the frame from end to end in the extended LAN.

Digital's LAN Bridge 100, LAN Bridge 150, and LAN Bridge 200 models are nontranslating type bridges, and preserve the CRC.

Digital's DECbridge 500/600 series units are translating type bridges, and as such, recalculate the CRC to reflect the altered information (refer to Section 2.1.3.2 for more information about the translation feature).

5.1.5 Low-Performance Controllers

Low-performance controllers can cause LAN problems as the amount of traffic on the LAN increases. Some low-performance controllers only provide a single buffer for receiving packets. Other packets immediately following the first (within the proper Physical layer timing parameter) are not able to be properly received.

The store-and-forward nature of the bridge changes the way frames are forwarded in the extended LAN. For example, two packets received with several packets between them (or some LAN idle time) on one bridge port may be sourced *back-to-back* on the outbound LAN. A single buffer controller would have trouble receiving the second packet.

Even for multibuffered controllers, an extended LAN may increase the processing needs or memory needs to ensure that the packets are correctly processed.

5.1.6 Frame Loss Caused by Congestion

A LAN may transmit frames from only one station at a time. Congestion occurs when a LAN is busy at the time another station (bridge) is ready to transmit a frame. A frame may be lost due to congestion for three reasons:

- If a frame reaches its lifetime in a bridge (2 seconds) before the bridge can transmit the frame, the bridge discards the frame.
- When the LAN becomes available, all bridges with frames in their outbound queues attempt to send them. If more than one station attempts to send a frame at the same time, a collision takes place. All stations then follow a prescribed procedure where they back off, then try again after waiting a random amount of time. If a bridge is unable to transmit the frame, it repeats this process. If the bridge experiences 16 collisions without sending the frame, it discards the frame. In this case, the outbound LAN is overloaded.
- If the bridge fills its transmit queue while waiting for the LAN to become available, the bridge begins to discard frames that it would otherwise be forwarding.

5.2 Bridge Performance

Bridges must perform several functions without adding unacceptable delay to the performance of the extended LAN:

- Recognize, process, and transmit spanning tree Hello messages.
- Recognize, process, and respond to management requests.
- Forward and filter frames.

This section discusses constraints on bridge performance of these functions.

5.2.1 Forwarding and Translating Process Module

The forwarding and translating process module performs these functions:

- Checks the destination address of the frame to determine whether to forward or filter the frame. The LAN Bridge 200 and DECbridge 500/600 series units also check the frame's source address.
- Checks the protocol type of the frame to determine whether to forward or filter the frame (LAN Bridge 200 and DECbridge 500/600 series).
- If the outbound port is connected to a dissimilar LAN, the (DECbridge 500/600 series) module modifies *(translates)* the frame's format to meet the requirements of the recipient LAN.
- If the frame received on the inbound port is an Internet Protocol frame, and the frame size is larger than that allowed on the outbound port, the (DECbridge 500/600 series) module fragments the frame into several smaller packets to accommodate the recipient LAN. Refer to Section 2.1.3 for more information about the fragmentation of IP frames.
- Checks both source and destination address of received frames for their disposition in the address database. If the source address is not listed, it is added (*learning*) to the address database along with its associated port address.

5.2.1.1 Discard Rate

The bridge must be able to dispatch frames quickly enough so that it is able to recognize, process, and dispatch Hello messages and management requests addressed to the bridge. The goal is to ensure that processing frames does not cause the bridge to become congested. This means that the bridge's discard rate should be faster than the worst-case frame arrival rate, or that Hello messages and management requests are split off and treated with higher priority.

5.2.1.2 Forwarding Rate

A bridge must be able to process frames as quickly as the access method for the corresponding LAN allows. As a bridge completes processing a frame to be forwarded, it sends it to the outbound port. If the LAN is free, the port dispatches the frame. If the LAN is not free, the port adds the frame to its transmit queue and waits for the LAN to become free. If the LAN is congested, frames may accumulate in the port's transmit queue. If the transmit queue becomes full and no frames in the queue have exceeded their frame lifetime, frames received and processed by the bridge are discarded until space becomes available on the transmit queue.

5.2.1.3 Forwarding Latency

Forwarding latency for a bridge is the delay caused by a bridge processing a frame. Processing time does not include the time that the frame waits in the queue to be transmitted on the outgoing data link. Forwarding latency is measured from the time the last bit of the frame is received to the time the first bit of the frame is transmitted, with the assumption that there is no outbound congestion.

5.2.1.4 Frame Lifetime

The frame lifetime in a bridge is the amount of time from the beginning of reception of the frame to the beginning of transmission of the frame. Frame lifetime includes the time that the frame waits in the queue to be transmitted on the outgoing data link.

Frames that have resided in the bridge for less than 1 second are not discarded unless outbound congestion is such that no transmit buffers are available. Frames that reside in the bridge for more than 2 seconds are discarded. As a result, the maximum lifetime of a frame in an extended LAN with a diameter of seven bridges is about 15 seconds. With a 2-second frame lifetime in a bridge and a maximum extended LAN diameter of seven bridges, it is impossible for a frame to reach its extended LAN 15-second lifetime.

5.2.2 Forwarding Database

The forwarding and translating process module accesses the address database of the forwarding database to perform the following functions:

- Add address entries during the learning process.
- Check addresses contained in incoming frames.

The forwarding and translating process module accesses the protocol database (featured in some bridges) of the forwarding database to perform the following function:

- Check protocol values contained in incoming frames.

5.2.2.1 Learning Rate

When a bridge is powered up, its ports remain in the PREFORWARDING state for an amount of time defined by the Forward Delay spanning tree parameter. During the second half of the forward delay, the bridge learns the location of station addresses by examining the frames that it receives. Bridges also learn station addresses while forwarding.

An important function of a bridge is to maintain a correct address database for efficient transmission of packets to the intended LANs. A bridge should have enough address learning capability to ensure that an up-to-date table is maintained at all times. Also, the address table should be large enough to ensure an address space for all of the active nodes on the network.

The bridge's learning and update rate must be sufficient to maintain these addresses *and* to ensure their correctness even during the times when the heaviest utilization of the attached LANs and the forwarding engine is taking place. For extended LAN stability, in fact, this is the most important time to ensure progress on address learning.

For example, if the address table has the incorrect port associated with an address, the bridge will forward the frame to the incorrect port and actually be responsible for causing excess traffic for the LANs and other bridges in the configuration. Alternatively, if the address is not listed in the address table, the bridge will *flood* the frame and potentially use excess bandwidth.

Ensuring forward progress of information updates to the bridge's address table is critical to the stability of the extended LAN.

5.2.2.2 Age Rate

Bridges use two different spanning tree parameters that govern the aging out of the inactive forwarding database address entries: the Forwarding Database Normal Aging Time and the Forwarding Database Short Aging Time. The Normal Aging Time parameter controls address aging during normal network operation; the Short Aging Time parameter controls address aging after a topology change.

Both parameter values can be set with network management and their values may affect network performance. Lower values may cause bridges to delete unreferenced (inactive) addresses more frequently, increasing network traffic because fewer packets may be filtered if the nodes are truly active. Higher values will cause bridges to remove unreferenced address entries less frequently. If there are more active nodes than the size of the database, then the forwarding database will reach its maximum and other (more active) nodes may not be included in the current active list.

New addresses cannot be stored in the filled forwarding database and network traffic might increase, since frames sent to these addresses could not be filtered.

5.3 Management

Bridges must be able to process management requests during high network utilization. Many situations arise where management, through the use of the inband signaling (same network path as the data being sent), would become unstable if the forwarding of packets took precedence over reacting to management directives to the bridge.

For example, a network manager attempting to track down the source of heavy network traffic might never learn the source of all the traffic if the bridge disregarded management directives and, instead, prioritized its resources to the forwarding of frames. A bridge responding in this fashion would eventually process the management directive *after* the event causing the heavy traffic (broadcast storm or *screaming* node, perhaps) had passed (obviously too late to be of any value to the network manager).

To ensure a stable and manageable extended LAN, the bridge must ensure that progress is made on management requests to the bridge management entity.

5.4 Spanning Tree

An argument similar to the one made for the guarantee of forward progress on the management process can also be made for the spanning tree entity. The extended LAN is only stable after the spanning tree algorithm has established that there are no data link loops and has included all of the LANs in the spanning tree.

During times of heavy traffic on the extended LANs, the bridge must maintain a correct and stable spanning tree. In fact, if the bridge cannot maintain the spanning tree in a stable state, the bridge itself might actually contribute to the added traffic.

For example, consider a case where the bridge exits the BACKUP state because the port failed to deliver the spanning tree messages from a better path for a designated bridge on the LANs. As the affected bridge enters the FORWARDING state, it creates more traffic by creating a data link loop with the other bridge(s).

The bridges processing priorities must ensure progress on spanning tree messages received from all ports.

6

Configuration

Digital's network bridges can be placed on an extended local area network and made to work directly as configured when shipped by the manufacturer. The default parameter values allow the bridges to work in most extended LAN environments.

Default bridge configurations, however, may not be the most efficient configurations for every network. In many cases, you can improve the efficiency and operation of the extended LAN by tailoring the network and configuring the bridges to your particular network application.

This chapter illustrates some of the potential problems in configuring extended LANs and provides guidelines to help you avoid or solve those problems and optimize the operation of your network.

6.1 Physical and Logical Topologies

Although the physical topology of an extended LAN can be arbitrary, the logical topology must be loop free so that messages can be forwarded correctly. This section reviews the spanning tree process that performs this function and summarizes the criteria used by the spanning tree algorithm to convert a physical topology to a logical, loop-free topology.

All bridges on the extended LAN continually perform the spanning tree process so that they can respond dynamically to changes in the network such as network or bridge failures or the addition of new bridges. As described in Chapter 3, the spanning tree algorithm selects a root bridge for the network, assigns a designated bridge for each LAN, and places bridges in the BACKUP state where loops are encountered. All bridges participate in the spanning tree process by passing information through Hello messages. Of the many parameters specified in each Hello message, the following two are important for determining the logical topology of the network:

Root Priority — This bridge parameter (value from 1 to 255) indicates the likelihood of each bridge becoming the root bridge of the network. Bridges shipped by Digital typically set the Root Priority to a default value of 128. However, this value can be changed with bridge management software. The lower the priority number, the higher the bridge's likelihood of becoming the root bridge.

Root Path Cost — The Root Path Cost is the sum of the line costs for each inlink between the bridge and the root bridge. A bridge computes its Root Path Cost by listening to Hello messages transmitted by neighboring bridges on the attached LANs. Each Hello message contains the distance to root of the transmitting bridge. The bridge calculates its own distance as the minimum path through any of its ports to the root bridge.

In addition to these two parameters, the bridge identification (bridge ID) value is an important factor in determining the logical topology. This bridge ID, which is assigned to the bridge during manufacturing, serves as a *tie-breaker* in the selection of the root bridge or the designated bridges on the network.

Briefly, the spanning tree process works as follows:

1. First, it elects the root bridge for the network. The bridge with the lowest priority number is selected as the root bridge. If multiple bridges have the same lowest priority number, the bridge with the lowest address becomes the root bridge.

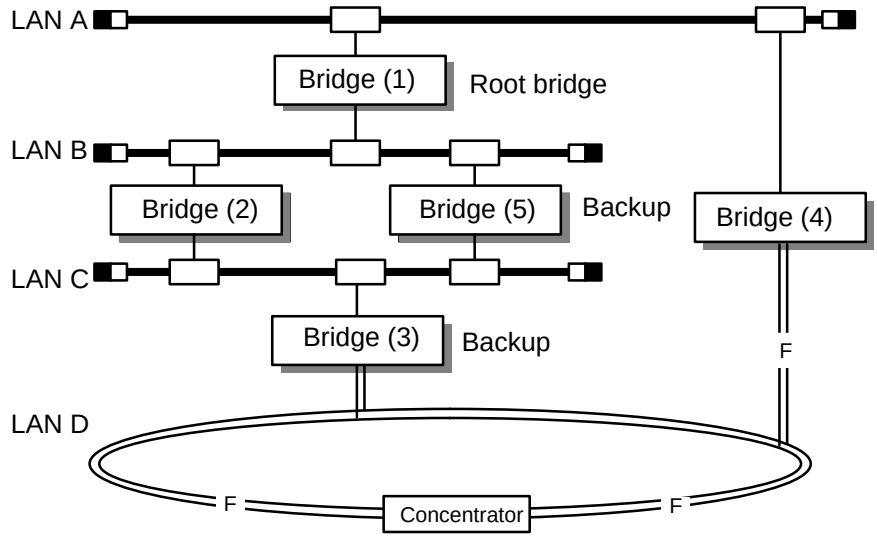
2. Next, the spanning tree process elects a **designated bridge** for each LAN. If a LAN is connected to more than one bridge, the bridge with the lowest Root Path Cost is selected as the designated bridge. If more than one bridge has the same Root Path Cost value, the bridge with the lowest node ID value becomes the designated bridge for that LAN.

3. The spanning tree process then places all bridges that are not part of the spanning tree (that is, any bridge that is not a root bridge or a designated bridge) in a backup mode of operation. While in the backup mode, a bridge still listens to Hello messages so that it can detect changes in the network and enter the FORWARDING state, if necessary.

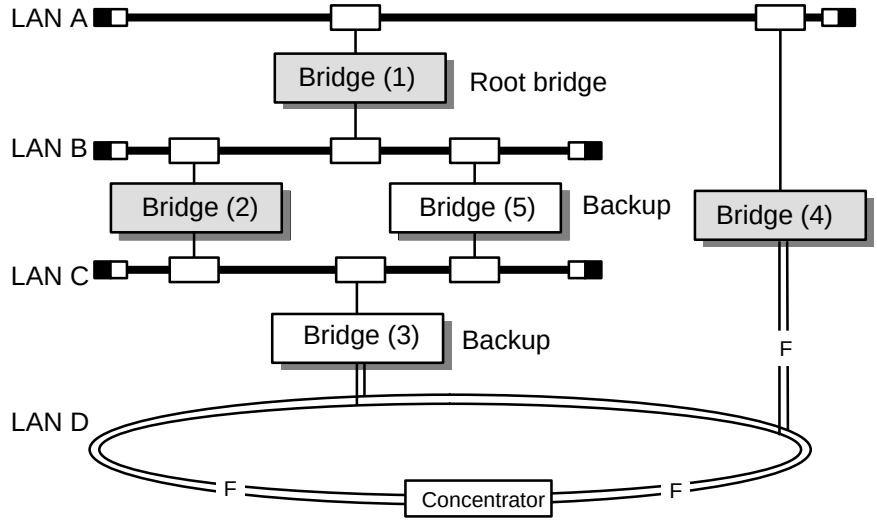
Figure 6–1 shows an example of a physical topology transformed to a logical topology by the spanning tree process, using the criteria described previously. For this example, assume that all bridges have the same Root Priority value (default of 128) and the same Line Cost value (default of 10). Numbers 1 through 5 represent the relative node ID values of the bridges (that is, bridge 1 has the lowest ID and bridge 5 has the highest ID).

Figure 6–1: Physical Topology Transformed to a Logical Topology

A. Physical Topology



B. Logical Topology



NOTE: Root priority and line cost values are assumed to be the same for all bridges. Numbers in parentheses indicate relative bridge ID values.

LKG–5399–90I

- Bridge 1 is selected as the root bridge of the spanning tree (based on its having the lowest bridge ID) and is also the designated bridge for LANs A and B.
- Bridge 2 is selected as the designated bridge for LAN C because, having the lower ID, it wins the tie-breaker with bridge 5 (both bridges have the same Line Cost).
- Bridge 4 is selected as the designated bridge for LAN D (because it has a lower Root Path Cost than bridge 3).
- Bridges 3 and 5 are set to backup mode of operation.

Since the spanning tree process is ongoing, any changes to the status of these bridges will result in a new logical topology. For example, in Figure 6–1,:

- If bridge 2 fails, bridge 5 will go into the FORWARDING state and become the designated bridge for LAN C.
- If bridge 4 fails, bridge 3 will go into the FORWARDING state and become the designated bridge for LAN D.
- If bridge 1 (the root bridge) fails, bridge 2 will become the new root bridge and the designated bridge for LANs B and C. Also, bridge 3 will become the designated bridge for LAN D and bridge 4 will become the designated bridge for LAN A.

6.2 Why Modify Bridge Configurations?

The spanning tree process, using default bridge parameters, creates a loop-free network topology that works. However, that topology may not be very efficient in your particular network environment. It may be advantageous to modify the bridge configurations to meet your needs.

With some forethought, you can anticipate the network configuration that the spanning tree process will produce (it is deterministic) and can specify bridge parameters to achieve a more desirable configuration. This section provides examples of some configuration problems that could occur and ways to avoid them.

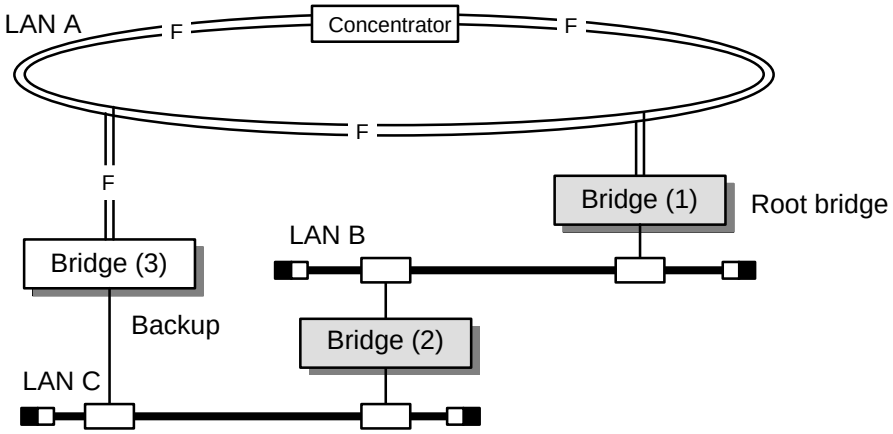
6.2.1 Example 1—Efficient Bridge Configurations

Problem

Consider the example with three bridges shown in Figure 6–2. For simplicity, assume that all three bridges have the same Root Priority value and the same Line Cost. Numbers 1, 2, and 3 represent the relative bridge ID values of the bridges. Based on these parameters, the spanning tree process configures the bridges as follows:

- Bridge 1 is the root bridge.
- Bridge 2 is the designated bridge for LAN C.
- Bridge 3 is set to the backup mode of operation.

Figure 6–2: Example 1—Sample Default Configuration



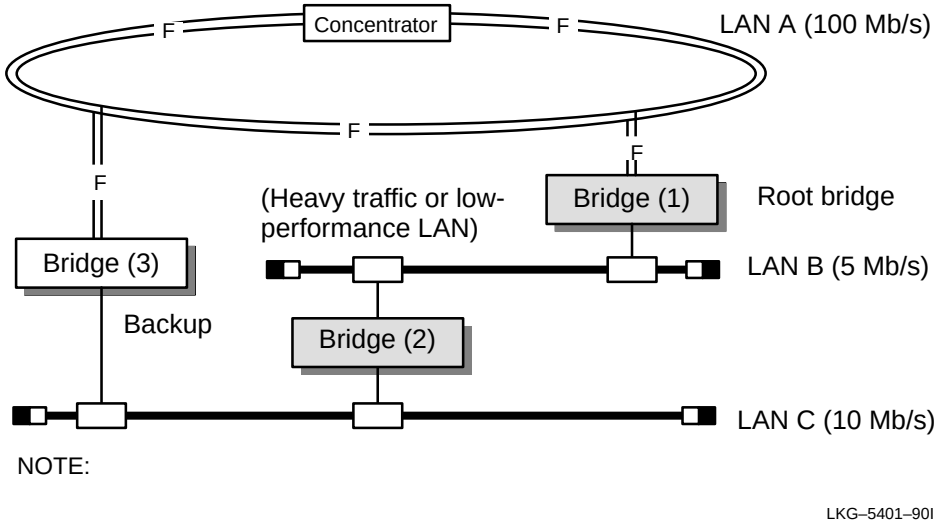
NOTE: Root priority and line cost values are assumed to be the same for all bridges.
Numbers in parentheses indicate relative bridge ID values.

LKG–5400–90I

The default configuration shown in Figure 6–2 works. All three LANs are interconnected and the network is loop free. However, depending on the particular characteristics and traffic loads of the three LANs, this configuration may not be very efficient. For example, suppose that LAN B is a lower-performance LAN than the other two LANs, as shown in Figure 6–3. All traffic between LAN A and LAN C has to pass through the slower LAN B, which is not a desirable path.

Alternatively, suppose that LAN B carries a very heavy traffic load of its own (for example, it could be the LAN used by an engineering department). This configuration would further increase that heavy load on LAN B by forwarding the LAN A-LAN C traffic through it (see Figure 6–3).

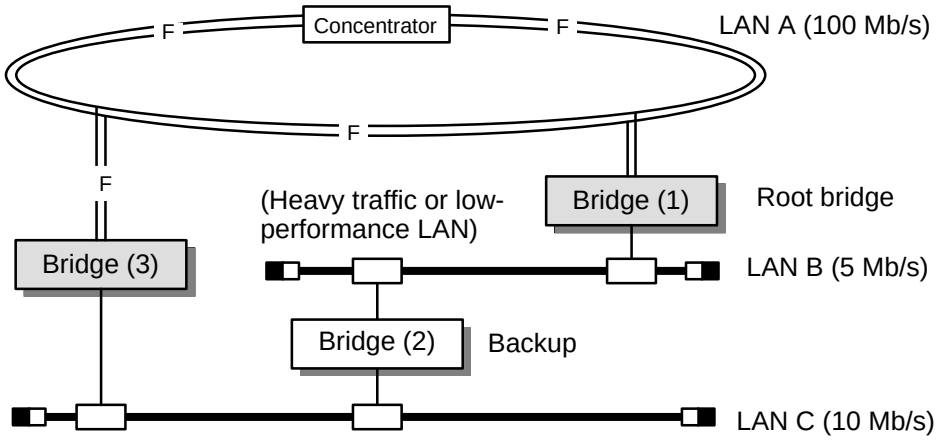
Figure 6–3: Example 1—Inefficient Bridge Configurations



Possible Solution

In both instances described in this example, a more efficient configuration is to establish bridge 3 (rather than bridge 2) as the designated bridge for LAN C, and to place bridge 2 in a backup mode of operation (see Figure 6–4). This places LAN B on the perimeter of the extended LAN, where it does not have to handle traffic from LAN A and LAN C. This network reconfiguration can be easily performed by using the bridge management software to make the Line Cost value of bridge 3 lower than that of bridge 2.

Figure 6–4: Example 1—A More Efficient Configuration



NOTE: Root priority values are assumed to be the same for all bridges.
Numbers in parentheses indicate relative bridge ID values.

LKG–5402–90I

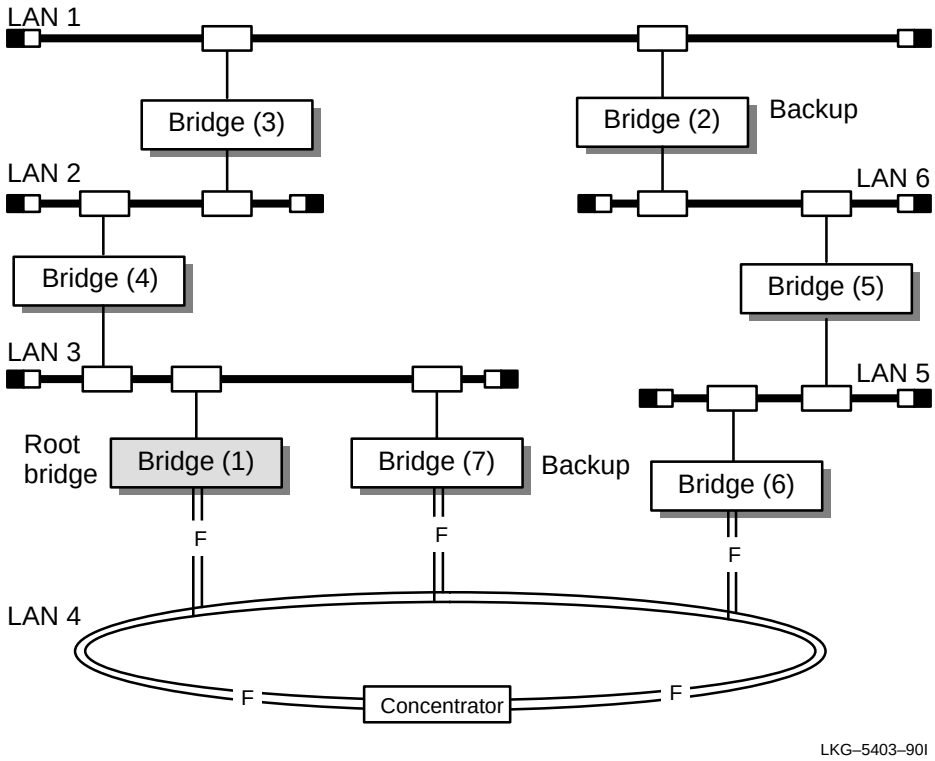
6.2.2 Example 2—Backup for Root Bridge

Problem

In the sample bridge configuration shown in Figure 6–5, assume that all the bridges have the same Root Priority and Line Cost values, and that the numbers 1 through 7 indicate their relative node ID values.

- Bridge 1 is selected as the root bridge.
- Bridge 7 is in the backup mode of operation.
- Bridge 2 is also in the backup mode of operation because it is not the designated bridge for either of the LANs it is connected to.
- Each of the other bridges is a designated bridge for an attached LAN.

Figure 6–5: Example 2—Backup for the Root Bridge



This configuration presents no particular problems; however, if the configuration changes (for example if bridge 1, the root bridge, fails), bridge 2 becomes the new root. If the LANs on both sides of bridge 2 carry a heavy traffic load, then all messages emanating from the new root will add to the already heavy traffic load on those LANs.

Possible Solution

Bridge 7 is a better backup for the root bridge. Therefore, you can configure the network so that bridge 7 is selected as the root if bridge 1 fails. You can make this change by using bridge management software to lower the root priority of bridge 1 and bridge 7.

6.3 How to Modify the Bridge Configurations

The most flexible method for modifying bridge configurations is through the bridge management software (as described in Chapter 4). This software allows you to modify bridge parameters such as Root Priority and Root Path Cost, which affect the logical topology of an extended LAN.

Using bridge management (see Figure 6–4), you can change the Root Path Cost value for bridge 3 so that it is selected as the designated bridge for LAN C, rather than bridge 2.

Without bridge management, the only way to reconfigure an extended LAN is to physically place bridges so that the node IDs cause the bridges to logically arrange themselves in the desired configuration. This is obviously not a very convenient method.

6.4 Considerations in Configuring the Network

When configuring an extended LAN, consider the following guidelines:

- Know the characteristics and traffic loads of the LANs so that you can place bridges appropriately to control the traffic. For example, group nodes that have heavy traffic loads together on a LAN segment and use a bridge to separate that segment from the rest of the network.

- Place LANs with heavy traffic loads on the perimeter of the extended LAN, if possible.
- Choose the network technology that is suitable to the needs and growth patterns of your business.
- Set up bridge parameters so that the resulting spanning tree configuration is most efficient for that particular network.
- Use the bridges' filtering features to help resolve potential traffic problems or security problems on the network.
- Establish the root bridge near the center of the topology.
- Consider what the new configuration will be if the root bridge or a designated bridge fails. (The spanning tree process is deterministic, allowing you to determine what the logical topology will be for any set of conditions.) Arrange the network topology so that any reconfiguration caused by a bridge failure will still allow proper network operation.
- If you change bridge parameters, consider how that change affects the rest of the network. For example, if you make changes to one bridge, also make corresponding changes to its backup bridge or other bridges.
- To ensure that the network link remains available, add a backup bridge. If the designated bridge fails, communications will not be affected.

6.5 Packet Filtering/Forwarding

Bridges are store-and-forward devices used for filtering network traffic. Each bridge has an address database that contains addresses of stations connected to both of its LANs. When the bridge detects a packet on either of its LANs, it checks the destination address of that packet against the addresses stored in its database.

If no filter matches exist (as described in subsequent paragraphs), and if that packet is addressed to a node on the other side of the bridge or to a destination address that is not in the database, the bridge forwards the packet. If the packet is addressed to a node on the same side of the bridge, the bridge filters the packet (it does not forward it to the other side).

Address information can enter the database in either of two ways:

1. The bridge *listens* to network traffic and acquires a working knowledge of which node is located on which of its two LANs. It acquires this knowledge by reading the source address of each incoming packet and by noting on which of the two LANs that source is located.

2. In addition to *learned* addresses, the address database can also receive entries from bridge management. By specifying a station address and the side of the bridge where that station is located, you can *lock* that station to that side.This creates a *permanent* address for that node and causes the bridge to ignore any learned information that differs from this permanent address. An example of how this feature may be used is given in the following section.

Destination address filtering decreases network traffic by keeping local traffic local and only forwarding packets that are destined for the opposite side of the bridge. All Digital bridges perform destination address filtering. The LAN Bridge 200 and DECbridge 500/600 series, however, also support source address filtering and protocol filtering, as described below.

Source address filtering — Using bridge management, you can prevent all packets emanating from a specified node from being forwarded to the other side of the bridge (regardless of their destination address). Source address filtering can also be used to *lock down* a specified source node to one side of the bridge. This protects against masquerading nodes, as described in Section 6.6.5.

Protocol filtering — Bridge management can also be used to prevent specified protocols from being forwarded across the bridge.

Both address and protocol filtering can be used in an inclusive or an exclusive fashion. For example, if you specify the Local Area Transport (LAT) protocol for protocol filtering, you can have the bridge filter only LAT protocol messages and forward all other protocols (if no other filters match), or you can have the bridge forward only LAT protocol messages (if no other filters match) and filter all others. Similarly, when you specify a set of node addresses for address filtering, you can have the bridge filter messages to those nodes and forward messages to all other nodes (if no other filters match), or you can have the bridge forward messages only to those nodes (if no other filters match) and filter messages to all other nodes (manual mode).

Note that filtering always takes precedence over forwarding. For example, if you specify that the bridge should always forward messages to a particular destination address (for example, node Z), and also specify that the bridge should always filter LAT protocols, any LAT protocol message destined for node Z will be filtered. In other words, the bridge will only forward a packet if no filters are set to filter that packet.

6.6 Packet-Forwarding Problems and Solutions

This section presents some examples of how the filtering features of bridges can be used to solve various extended LAN problems.

6.6.1 Example 1—Heavy Broadcast Traffic

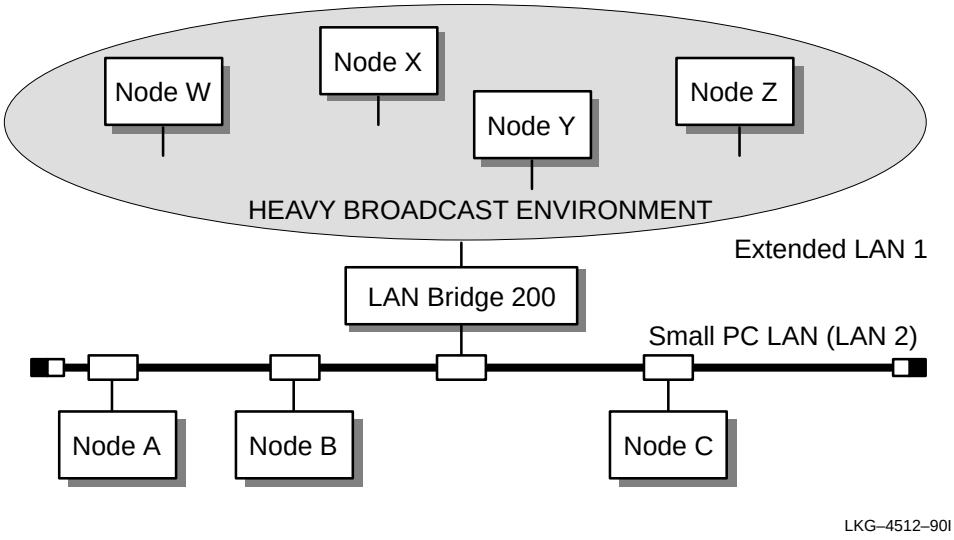
Problem

The extended LAN shown in Figure 6–6 illustrates some of the problems and possible solutions associated with heavy broadcast environments. Assume that the following network conditions exist:

- Nodes W, X, Y, and Z are only a few of many nodes on a large extended LAN (LAN 1) that handles heavy broadcast traffic. Much of the traffic on that LAN is due to the TCP/IP protocol used by many of the nodes.
- Nodes A, B, and C are small systems connected to their own LAN segment (LAN 2). They cannot handle a heavy traffic load but still must be able to communicate with certain TCP/IP nodes on LAN 1.
- A LAN Bridge 200 interconnects the two LANs.

The TCP/IP protocol can create a heavy traffic load because of the way it operates. When a TCP/IP node needs to request a service from another TCP/IP node but does not know the address of that node, the following action occurs:

Figure 6–6: Example 1—Heavy Broadcast Traffic



- Since the TCP/IP protocol does not use multicast addresses in communicating with other nodes, the requesting node sends an Address Resolution Packet (ARP) to the broadcast address. This transmits a message to all nodes on the network requesting the desired service.
- All nodes receive the ARP message and examine it to determine how they should react. Any TCP/IP node that can provide the service responds by informing the requesting node of that fact.
- Nodes that do not use TCP/IP, however, also examine the message and send it to a host which checks it and decides to disregard it. This consumes controller time, CPU time, and buffer space.
- This heavy broadcast traffic can become even worse if certain problems or error conditions occur. Also, some workstations can behave like routers and respond by appending a message to the transmitted message, thus increasing the already heavy traffic load.

As a result, in certain situations, a protocol such as TCP/IP can cause extremely heavy traffic loads on a network.

Possible Solutions

1. The address filtering feature of the LAN Bridge 200 and DECbridge 500/600 series provides a convenient way of handling the heavy traffic problem. To keep the traffic load on LAN 2 manageable, first determine precisely which nodes on LAN 1 must be able to communicate with the nodes on LAN 2. Then use address filtering to forward only messages between those selected nodes on LAN 1 and LAN 2, and filter all the rest.

   For example, suppose you determine that the nodes on LAN 2 need only communicate with nodes W, X, Y, and Z on LAN 1. You can use bridge management to place the bridge in manual mode. This clears the existing address database in the bridge, prevents the bridge from learning new addresses, and forces the bridge to accept only addresses supplied by bridge management. You can then specify the addresses of nodes A, B, C, W, X, Y, and Z, set up the bridge to only forward messages between those two groups of nodes, and filter all the rest.

   Alternatively, suppose you determine that nodes W, X, Y, and Z are the only nodes on LAN 2 that nodes A, B, and C need not communicate with. You can use the normal mode of the bridge, and set nodes W, X, Y, and Z to be filtered. That is, direct the bridge to filter all traffic sourced from or destined to nodes W, X, Y, and Z.

2. In certain cases, LAN 1 traffic may also be reduced by using the protocol filtering capabilities of the bridge (a LAN Bridge 200 in this example). Suppose the nodes on LAN 2 communicate with each other exclusively in a protocol such as AppleTalk. If none of the nodes on LAN 1 uses AppleTalk, then none of the AppleTalk traffic on LAN 2 needs to be forwarded to LAN 1. Protocol filtering will let you set up the bridge to filter all AppleTalk messages.

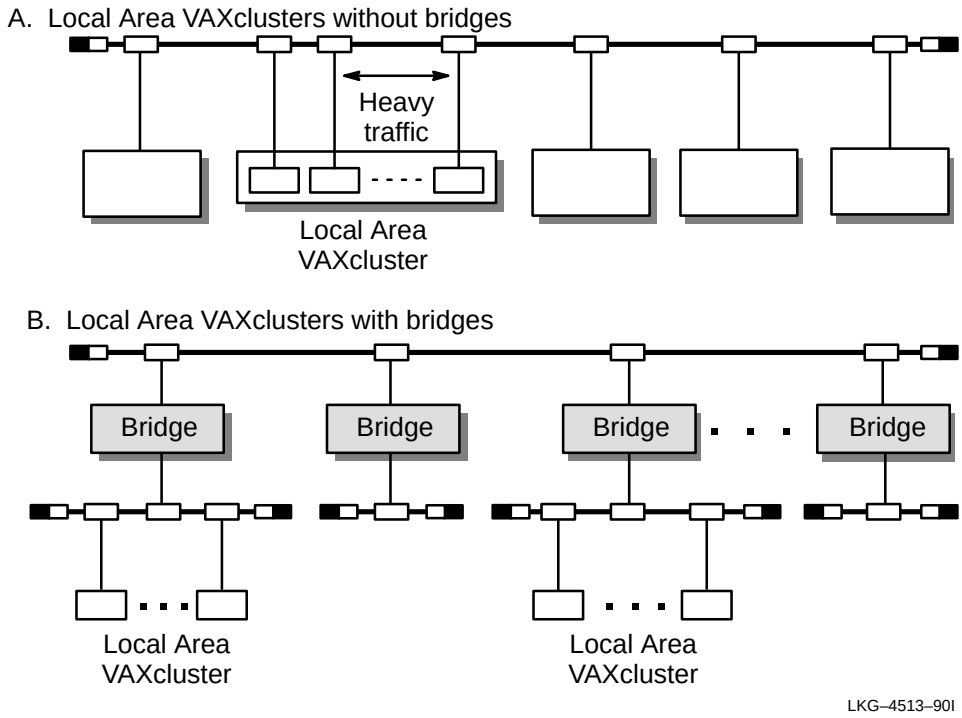
   If the bridge used in this example were not a LAN Bridge 200, then the only way to limit the heavy TCP/IP traffic on LAN 2 would be to physically group all nodes according to protocol. If possible, you would have to group all TCP/IP nodes together on the same LAN (LAN 1). Then, you could use destination address filtering to prevent the bridge from forwarding any messages directed to the broadcast address. This would prevent all broadcast messages on LAN 2 from being forwarded to LAN 1.

6.6.2 Example 2—Local Area VAXclusters

Problem

In a local area VAXcluster (LAVC) systems, as shown in Figure 6–7, all nodes in the cluster share the data resources of a central station. Therefore, by its nature, an LAVC may generate a heavy traffic load on the network that contains the cluster. This heavy traffic can needlessly bog down other network nodes that are not a part of the cluster, especially lower-performance nodes or controllers.

Figure 6–7: Example 2—Local Area VAXclusters



Possible Solutions

Cluster traffic can be isolated by grouping all the nodes of one cluster together on the same LAN and using a bridge to keep that cluster traffic away from other nodes (and other clusters). Figure 6–7 shows how bridges can be used to connect clusters to a network backbone while keeping them isolated from each other and from other nodes in the network.

If the bridges used are the LAN Bridge 200 or the DECbridge 500/600 series, the simplest way of isolating the clusters is through protocol filtering. Using bridge management, simply set each bridge to filter the LAVC protocol. This will keep the LAVC traffic local within each cluster and off the system backbone.

If other bridges without protocol filtering are used, it may be possible to employ address filtering to isolate each LAVC from the network backbone. Standard destination address filtering filters all cluster messages directed to physical addresses. However, all messages directed to multicast addresses are normally forwarded. To overcome this, you can determine the multicast address used by the cluster and add that address (or addresses) to the bridge's database. Then, through its normal destination address filtering, the bridge will prevent multicast messages from being forwarded to the network backbone.

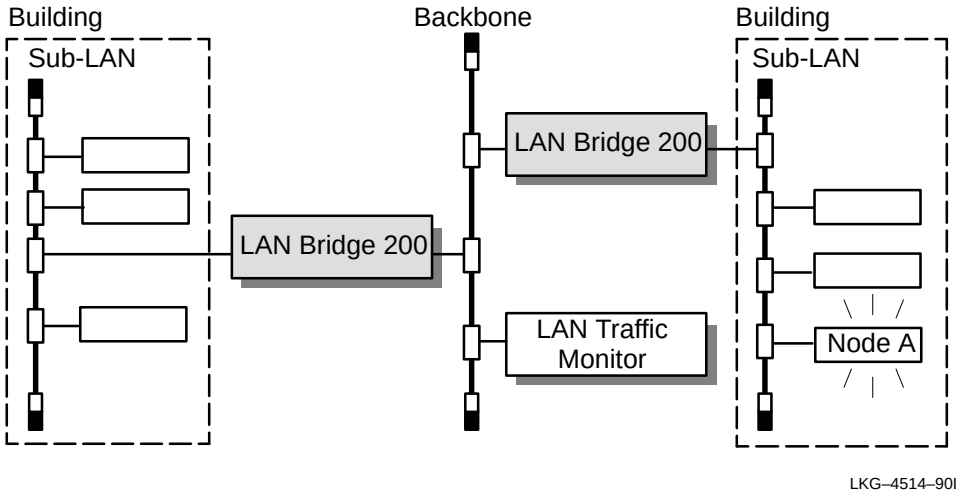
6.6.3 Example 3—Totally Blocking Out a Node

Problem

In some instances, you may find it useful to totally block out one or more nodes from the network backbone. Figure 6–8 shows one such example. That sample network consists of a backbone and several sub-LANs located in separate buildings to which you may not have access. Suppose node A on one of those sub-LANs malfunctions and repeatedly transmits messages that disrupt network operation. (Such a node is sometimes called a *screaming node* or *babbling node*.) Because you do not have access to the building, you cannot physically remove that faulty node from the network.

A similar situation can also exist when debugging new equipment or new protocols in a development environment.

Figure 6–8: "Screaming Node" Example



Possible Solution

By means of bridge management, you can use the address-filtering capabilities of the LAN Bridge 200, for example, to totally block all traffic to and from the defective node. This removes that node from the network.

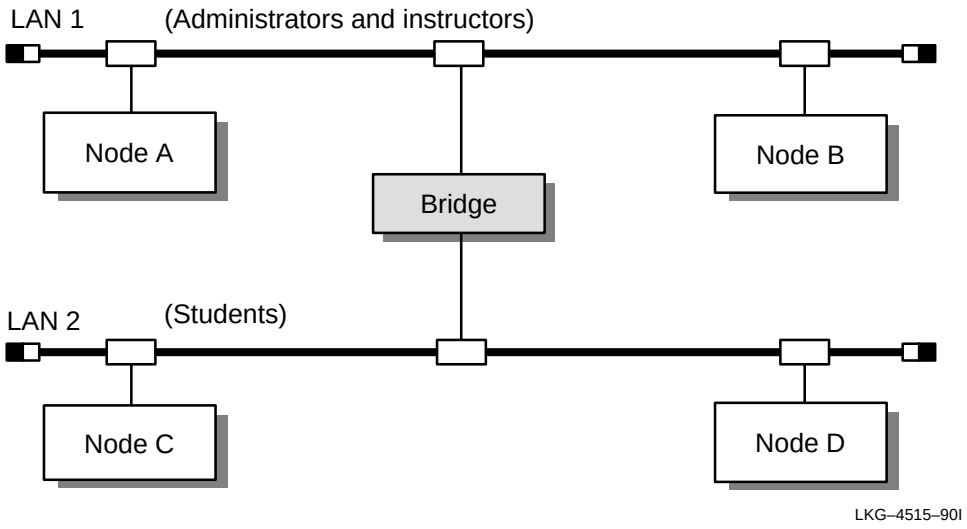
6.6.4 Example 4—Limiting Access to Nodes

Problem

In some environments, some nodes must be denied access to certain LANs or to certain nodes on a LAN. For example, in an academic environment, students cannot be allowed to access nodes that contain grading and other confidential information. In the example shown in Figure 6–9, all the nodes used by instructors and administrators are connected to LAN 1 and all the students' nodes are connected to LAN 2.

Assume the grades are stored on node A. Node B is used by instructors to specify or modify the grades and, therefore, must be able to communicate with node A. Node C, however, is a students' node and definitely should not have access to node A.

Figure 6–9: Extended LAN in an Academic Environment



Possible Solutions

1. If the bridge is a LAN Bridge 200 or a DECbridge 500/600, source address filtering can be used. Using bridge management, you can send the address of node C to the bridge and specify that all messages emanating from that node should be filtered. This will prevent node C from accessing any of the nodes on LAN 1.

2. If the bridge does not have source address filtering, destination address filtering can be used (in a subtle way) to attain the same results. Using bridge management, you can purposely misinform the bridge that node A is connected to line 2 (rather than to line 1). This logically moves node A from line 1 to line 2 (in the bridge's address database), even though node A remains physically connected to line 1.

   Because nodes A and B are both physically on LAN 1, the bridge has no effect on the intercommunication between those two nodes. However, when node C tries to communicate with node A, the bridge does not forward those messages because it thinks that both node A and node C are on LAN 2. As a result, all packets from node C remain on LAN 2 and are never received by node A (because it is physically connected to LAN 1).

NOTE

In the above example, node A is intentionally *locked down* to the wrong line to protect against malicious attacks. Use this approach with care because the locked-down node will become inaccessible to certain other nodes, thereby becoming isolated from parts of the network.

6.6.5 Example 5—Masquerading Nodes

Problem

A *masquerading node* is a node that is physically located on one side of the bridge, posing as a node on the other side of the bridge. In the previous example, the goal was to prevent node C (the students' node) from accessing node A (the grades). Even if the solutions proposed in that example were implemented, a resourceful student on node C might try to gain access to node A by masquerading as node B (by using node B's address as its own source address). This would defeat the source address filtering of node C and cause the bridge to forward the message to node A, thus giving node C (the masquerading node) access to node A (the grades).

Possible Solutions

Using bridge management, you can lock down node B to line 1. Then, if the bridge receives a message on line 2 from a node claiming to be node B, it filters the message.

6.7 Multiport Filtering/Forwarding

The network manager can restrict access to the various extended LANs by controlling the disposition of the ports on each bridge. In general, the following dispositions can be applied to any port:

`SET node x PORT=n` – Lock down node x to port n.

`SET DISPOSITION=FILTER` – Always filter.

`SET DISPOSITION=FORWARD` – Always forward (2–port bridges only).

NOTE

The command formats used in this section are for illustration purposes only. To determine the exact command formats for your network management system, refer to the documentation provided with that system.

The FORWARD disposition does not apply to multiport bridges because it is not realistic to forward packets with physical addresses to all ports. Packets directed to physical addresses are forwarded only to the LAN containing that address.

On multiport bridges such as the DECbridge 600 series, similar dispositions can be applied to individual ports or groups of ports. For example,

`FILTER LINE 1` – Always filter to nodes on port 1.

`FORWARD LINE 2,3,4` – Only forward to nodes on ports 2, 3, and 4.

Note that these two dispositions accomplish the same goal. Messages can only be forwarded to ports 2, 3, and 4; never to port 1.

On multiport bridges, network management can also assign dispositions to individual input/output port pairs, for a particular address or protocol. For example:

```
INPUT PORT (1), OUTPUT PORT (2)
INPUT PORT (2), OUTPUT PORT (1,3)
INPUT PORT (3), OUTPUT PORT (4)
INPUT PORT (4), OUTPUT PORT (0)
```

allows forwarding of a specified address or protocol from port 1 to port 2, port 2 to ports 1 and 3, and port 3 to port 4. No forwarding of that address or protocol is allowed from port 4 to any other port.

Note that the bridge's learning process overrides entries in the forward/filter map. For example, if the bridge learns that an address originates on line 1, it will not unnecessarily forward packets destined to that address and already on line 1 to other ports on the bridge, even if the forward/filter map allows the bridge to do so.

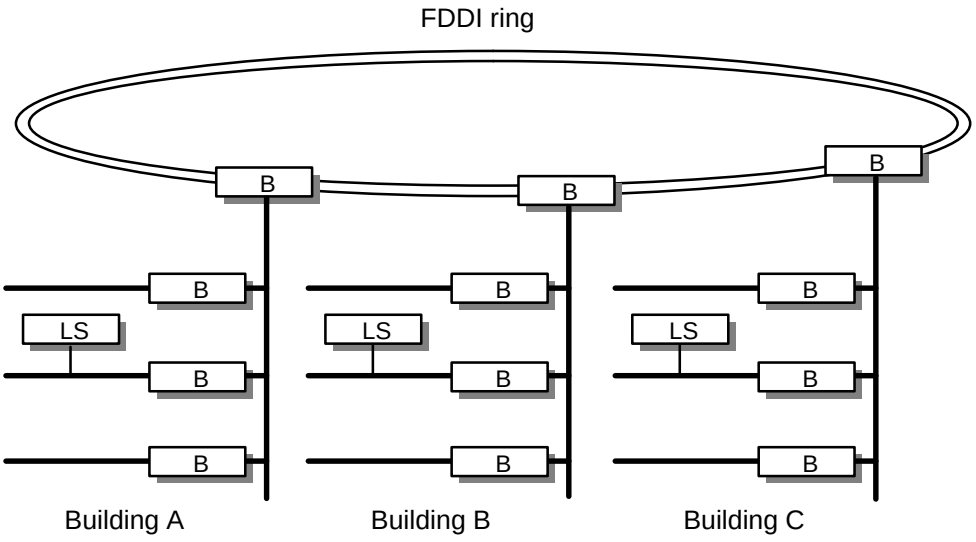
The ability to control individual port pairs greatly increases the flexibility of multiport bridges. Management, however, should use only the level of filtering/forwarding control that is needed to do the job. Applying a more detailed level of filtering than is needed unnecessarily uses more processing time.

6.7.1 Example 6 – Using Multiport Bridges to Control Load Server Traffic

Problem

In the extended LAN shown in Figure 6–10, nodes on any of the LAN segments can request load service at any time. Following a general power-off condition, all nodes may request load service at the same time, flooding the network, and preventing or delaying the load servers from servicing the requests.

Figure 6–10: Example 6 – Extended LAN With Load Servers



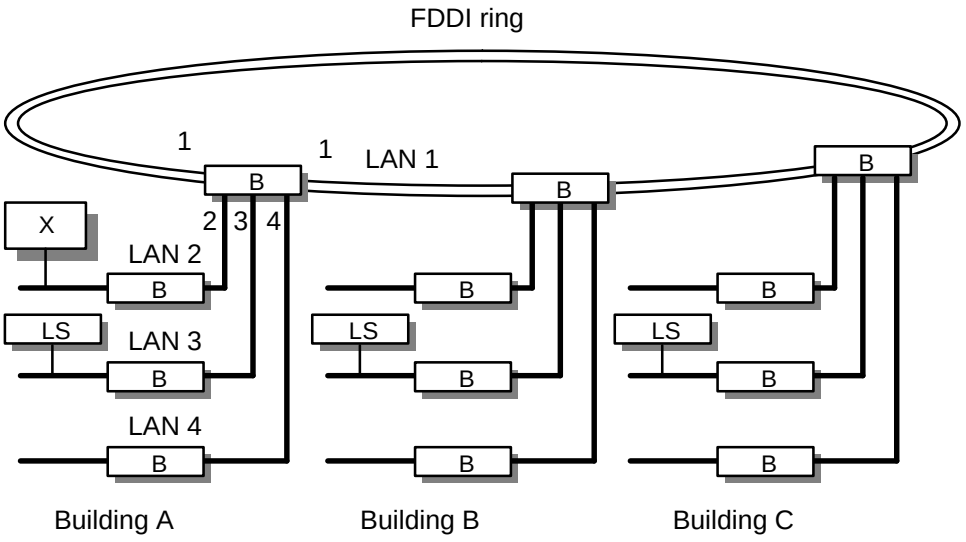
Legend:
B – Bridge
LS – Load Server

LKG–5404–91I

Possible Solution

Any DECbridge 600 series multiport bridge can be used to control the direction and the flow of traffic between the load servers and the nodes. The new configuration is shown in Figure 6–11.

Figure 6–11: Example 6 – Using Multiport Bridges to Control Load Server Traffic



Legend:
B – Bridge
LS – Load Server
X – Node

LKG–5405–91I

Network management sets the multicast load request address to a FORWARD disposition, and specifies which ports will be allowed to forward load request packets. In this example, the following per-port *forward map* would be defined for each of the three bridges on the ring:

```
INPUT PORT (1), OUTPUT PORT (0)
INPUT PORT (2), OUTPUT PORT (3)
INPUT PORT (3), OUTPUT PORT (0)
INPUT PORT (4), OUTPUT PORT (3)
```

NOTE

Refer to the documentation provided with your network management software for specific command formatting information.

With these port assignments in building A, load requests on LAN 2 will go only to LAN 3, load requests on LAN 4 will go only to LAN 3, and load requests from LANs 1 and 3 will always be filtered.

For example, load requests from node X will go only to LAN 3, where the load server is located. In response to the load request, the load server will send its data to the physical address of node X. The bridge, through its address data base, knows that node X is located on LAN2. LANs 1 and 4 will never see the the load request or the load server data, thus limiting load server traffic to only those LAN segments involved in the transaction.

6.7.2 Example 7 – Using Multiport Bridges to Control LAVC Traffic

Problem

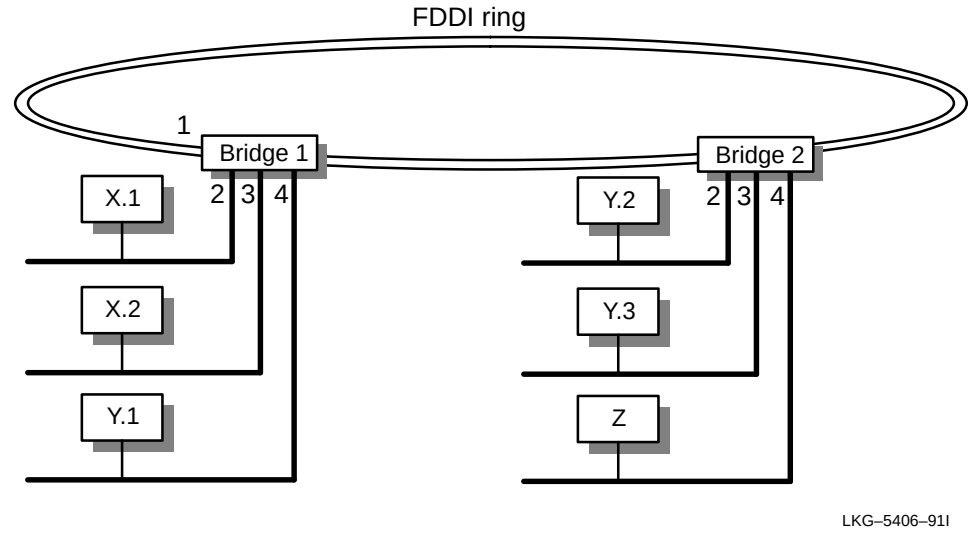
To isolate a local area VAXcluster (LAVC) from the rest of the network, you would typically group all the LAVC stations together on the same LAN segment (if possible), and use a bridge to filter that LAVC traffic from the rest of the network. As described in Example 2 earlier in this Chapter, a two-port bridge would filter the cluster traffic based either on its LAVC protocol or on its LAVC multicast address.

However, if all the stations in a cluster cannot be grouped on the same LAN, this filtering would prevent cluster stations on one LAN from communicating with those on another LAN.

Possible Solution

DECbridge 600 series multiport bridges can be used as shown in Figure 6–12. Nodes X.1 and X.2 represent stations belonging to the same cluster (X) but located on different LAN segments. Similarly, nodes Y.1, Y.2, and Y.3 are stations on cluster Y, but located on different LAN segments. All stations in cluster Z are located on the same LAN segment.

Figure 6–12: Example 7 – Using Multiport Bridges to Control LAVC Traffic



Network management can set the disposition in each bridge to forward LAVC protocols based on the following per-port forward map:

Bridge 1:

```
INPUT PORT (1), OUTPUT PORT (4)
INPUT PORT (2), OUTPUT PORT (3)
INPUT PORT (3), OUTPUT PORT (2)
INPUT PORT (4), OUTPUT PORT (1)
```

This allows LAVC traffic on cluster X to be forwarded among the X.1nodes (port 2) and X.2 nodes (port 3), but isolates it from the FDDI network and from the Y cluster on port 4.

Bridge 2:

```
INPUT PORT (1), OUTPUT PORT (2,3)
INPUT PORT (2), OUTPUT PORT (1,3)
INPUT PORT (3), OUTPUT PORT (1,2)
INPUT PORT (4), OUTPUT PORT (0)
```

This allows LAVC traffic on cluster Y to be forwarded among the Y.1nodes (through port 1 and the FDDI ring), the Y.2 nodes (port 2), and the Y.3 nodes (port 3), but isolates it from cluster Z on port 4. LAVC traffic on cluster Z is never forwarded to any other port, and remains isolated on that LAN.

6.8 Controlling Access to Bridges (Bridge Security)

There are two ways to control access to bridges:

- **Hardware switches** — Digital bridges contain two hardware switches (Port Access switches) that can prevent stations on that port from changing any of the bridge's internal parameters through the use of bridge management.

  When one of those switches is on, access to that port is enabled and any station with bridge management on that LAN can monitor and change bridge parameters. When the switch is off, stations on that LAN can read but cannot write bridge management parameters.

  These hardware switches also prevent downline loading and remote resetting of the bridge.

In Examples 4 and 5 of Section 6.6, the access switch on the student side could be turned off (disabled), and the access switch on the administrator/instructor side could be turned on (enabled). This would prevent stations on the student's LAN from changing bridge parameters.

- **Password** — The purpose of the password feature is to allow only authorized users to set and change bridge parameters. The password can be up to 16 characters long and is case-insensitive.

  To help protect against unauthorized access to the network, a 6-byte counter counts invalid password entries. If you suspect malicious attempts on the network, checking this counter regularly will help you detect those attempts.

6.9 Simple Bridges

High-performance bridges are store-and-forward devices that filter any packet destined for a node on the same port, or any packet that matches a management-set filter. All local packets are filtered and remain local.

Simple bridges are also store-and-forward devices, but some may not have the capability to perform any filtering. All packets received on one side of a simple bridge are forwarded to the other side, regardless of where the destination node is located. Also, simple bridges contain no spanning tree algorithm and therefore cannot be placed in a backup mode of operation. When powered up and operational, simple bridges are always in a FORWARDING state.

In reference to the bridge model described in Chapter 2, simple bridges can be considered as having no Spanning Tree entity, no protocol or address database, or possibly neither of the two.

Because of this inability to control simple bridges, avoid placing them in parallel with high-performance bridges. Such a configuration would cause the high-performance bridge to be in a backup mode as long as the simple bridge is operational, thereby wasting the filtering features of the high-performance bridge.

If placed in such a configuration, however, the high-performance bridge will continue to monitor the operation of the simple bridge by listening to its own Hello messages. If the simple bridge becomes inoperable, the high-performance bridge will take over and go into the FORWARDING state.

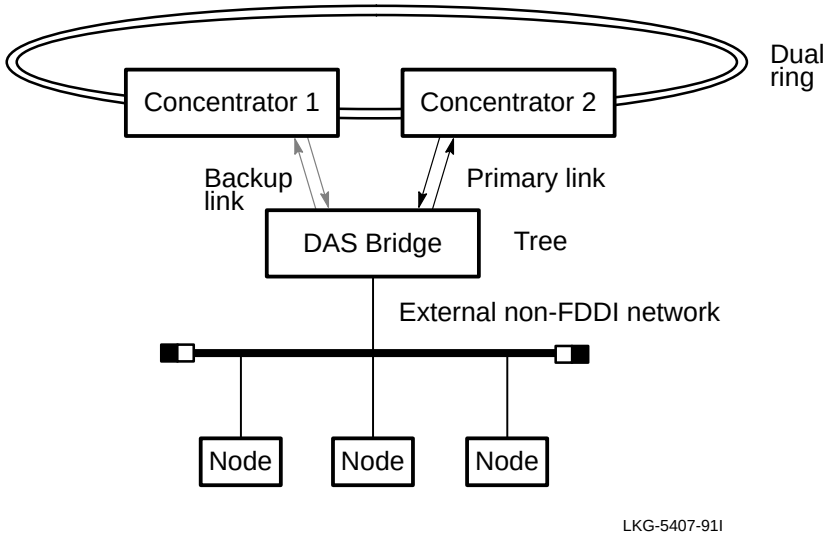
6.10 FDDI Bridges

The DECbridge 500/600 series are FDDI-to-802.3/Ethernet bridges. They act as a link between an FDDI ring and up to three IEEE 802.3/Ethernet ports. This series of bridges contain both single attachment station (SAS) and dual attachment station (DAS) options, depending on the individual models. SAS bridges connect to the FDDI dual ring through a concentrator. DAS bridges can connect to the dual ring directly.

6.10.1 Dual Homing

DAS bridges can also be connected to the FDDI ring through a dual homing configuration. Figure 6–13 shows an example of dual homing using two concentrators and a DAS bridge. One link of the DAS bridge is used as the primary link, and the other link is used as a backup link. If concentrator 2 or the primary link to the DAS bridge fails, the backup link to the DAS bridge, through concentrator 1, is activated. This helps to ensure uninterrupted service between FDDI and non-FDDI LANs.

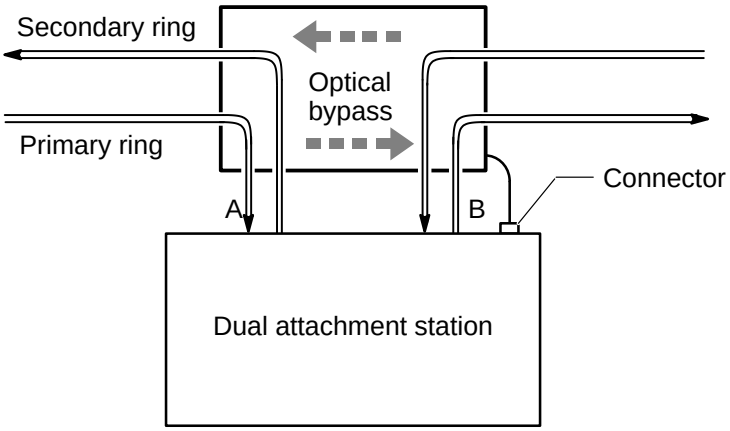
Figure 6–13: Example of a Dual Homing Topology



6.10.2 Optical Bypass Relay

DECbridge 500/600 series bridges also have an optical bypass relay connector (shown in Figure 6–14) that provides signals for an optical bypass relay switch. The switch can be used to maintain connectivity of the FDDI ring during a power failure or fault condition in the bridge. This switch allows the light to bypass the optical receiver in the bridge, thereby maintaining the operation of the FDDI ring.

Figure 6–14: Optical Bypass Relay



Optical bypass relays have a power penalty, however, which may cause the maximum allowable loss between stations to be exceeded. This limits the number of serially connected relays in the ring.

Other considerations when using optical bypass relays include:

- Bypass relays introduce additional loss in the network, and they do not perform repeater functions of amplifying and restoring the bit stream.
- By bypassing a station, the new distance between adjacent stations may exceed the maximum allowable value.
- Bypass relays, as any mechanical devices, may introduce less than reliable service to the network.

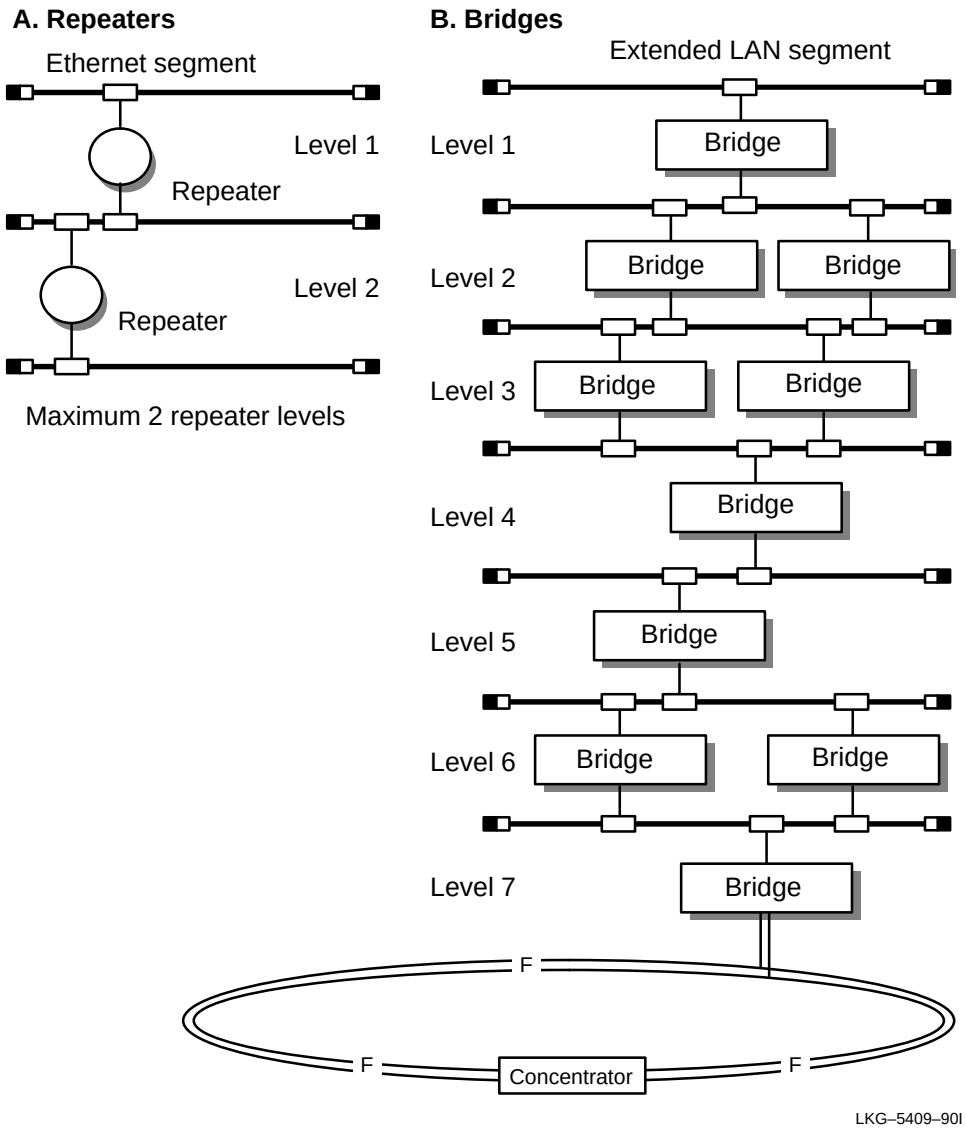
6.11 Bridges and Repeaters

An Ethernet repeater is used to interconnect multiple segments of standard Ethernet cable. As shown in Figure 6–15 (A), up to two levels of repeaters can be used to extend the length of a standard Ethernet LAN from 500 meters (1640 feet) for a single segment to 1.5 kilometers (0.93 mile) for three segments in series. Link segments may also be added for further extension. The repeater retimes, amplifies, and repeats the signal that it receives on one Ethernet segment and passes that signal to the next segment.

Digital bridges, however, store all received packets and, if necessary, regenerate and forward the packets to the opposite LAN. Bridges therefore enable you to build extended LANs many times larger than Ethernet single-LAN guidelines allow. Each bridge starts a new single-LAN segment, which itself can be extended by the use of repeaters, as described above. As shown in Figure 6–15 (B), up to seven levels of bridges can be used.

Another important advantage of a bridge over a repeater is the bridge's ability to isolate errors. Because the bridge looks at the entire frame before forwarding it, it can filter collision fragments, frames with errors, and other Physical layer faults. The repeater, with its bit store-and-forward ability, generally cannot isolate these errors.

Figure 6–15: Maximum Levels of Bridges and Repeaters



6.12 Spanning Tree Modes

This section describes the dual spanning tree modes that can be used by Digital's bridges.

6.12.1 Types of Spanning Tree Modes

The spanning tree computation process developed by Digital Equipment Corporation was first implemented in Digital's LAN Bridge 100 product. This spanning tree algorithm was offered to the IEEE and is now part of the IEEE 802.1d, MAC Bridge Standard.

Although the two algorithms are identical and produce exactly the same spanning tree topologies, the IEEE 802.1d standard uses several different parameters to implement the algorithm. For example, the multicast address used for bridge Hello messages in the LAN Bridge 100 spanning tree mode is different from the multicast address used for Hello messages in the IEEE 802.1 spanning tree mode. As a result, bridges that operate in one of these two spanning tree modes cannot understand Hello messages from bridges that operate in the other spanning tree mode.

To illustrate the problems involved in having the two types of bridges in the same network, assume that a LAN Bridge 100 is connected in a loop with a bridge using the IEEE 802.1 spanning tree mode. Because the two bridges cannot understand each other's Hello messages, the first one to receive a Hello message from the other will simply ignore the message and pass it on. The sender of that Hello message, detecting its own Hello message on both of its ports, will place itself in the BACKUP state. In this situation, the spanning tree algorithm is no longer deterministic. A user cannot predict (or control) which of the two bridges will be the forwarding bridge and which will be the backup bridge.

Another problem can arise if both bridges come up at the same time. That would cause both bridges to cycle continuously between the BACKUP state and the FORWARDING state.

6.12.2 Migration Bridges

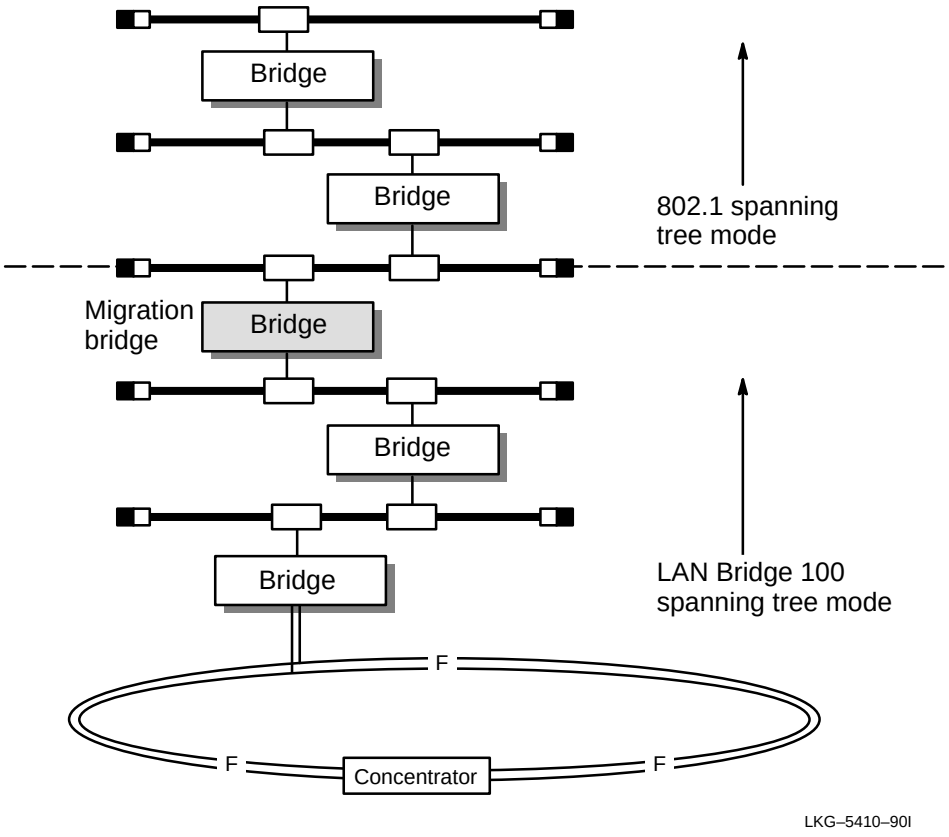
To help handle the problem in which networks have both types of bridges, some using the LAN Bridge 100 spanning tree mode and others using the IEEE 802.1 spanning tree mode, Digital manufactures the LAN Bridge 150, LAN Bridge 200, and DECbridge 500/600 series. These bridges can dynamically adapt to either the LAN Bridge 100 spanning tree mode or the IEEE 802.1 spanning tree mode, depending on what the network configuration requires. An Auto-Select software switch controlled by bridge management lets you either enable this *auto-select* feature, or lock the bridge in the IEEE 802.1 spanning tree mode. Bridges with this auto-select feature are sometimes called **migration** bridges.

When a migration bridge is installed, it defaults to the IEEE 802.1 spanning tree mode. If it detects any bridges operating in the LAN Bridge 100 spanning tree mode, it automatically switches to that mode and discards any Hello messages received from an IEEE 802.1 spanning tree bridge. This subdivides the network into two parts: a subnetwork that uses the LAN Bridge 100 spanning tree mode on one side of the bridge and a subnetwork that uses the IEEE 802.1 spanning tree mode on the other side (see Figure 6–16). Each of these two subnetworks operates in its own spanning tree mode and, more importantly, the resulting spanning tree topology is deterministic. If the migration bridge stops hearing LAN Bridge 100 spanning tree mode messages, it automatically reverts to the IEEE 802.1 spanning tree mode.

If the root bridge is a migration bridge, it would not normally be aware of the LAN Bridge 100 turning off (or failing), since root bridges transmit but do not receive Hello messages from other bridges. To overcome this problem, every 5 minutes the migration root bridge polls a LAN Bridge 100 that it knows exists on the network.

If the root bridge does not receive a response from the polled LAN Bridge 100 within a certain time, it assumes that the bridge is no longer operational. The root bridge then uses the multicast address to determine whether there is any other LAN Bridge 100 on the network. If there is another LAN Bridge 100, the root bridge notes its address and polls that bridge periodically to check on its operational status. However, if the root bridge does not detect any other LAN Bridge 100 on the network, it reverts to the IEEE 802.1 spanning tree mode.

Figure 6–16: Using a Migration Bridge in a Network



6.12.3 Special Application for a Migration Bridge

A migration bridge operating in the auto-select mode can be a useful tool for locating and replacing bridges that use the LAN Bridge 100 spanning tree mode. For example, assume that an existing network contains some bridges that use the LAN Bridge 100 spanning tree mode and other bridges that use the IEEE 802.1 spanning tree mode. Suppose that you want to locate and upgrade all the bridges that use the LAN Bridge 100 spanning tree mode with bridges that use the IEEE 802.1 spanning tree mode. A network manager could proceed as follows:

1. Upgrade all bridges that are known not to run a spanning tree implementation that complies with the IEEE 802.1d, standard.
2. Use a migration bridge as the root bridge for the extended LAN.
3. Use bridge management to determine whether any bridge is being polled by the root bridge. If a bridge is being polled, that bridge is operating in LAN Bridge 100 spanning tree mode and is keeping the root bridge in LAN Bridge 100 spanning tree mode. It is a bridge that was overlooked in step 1.
4. Upgrade that bridge to an 802.1 spanning tree bridge.
5. Repeat the procedure until the root bridge automatically switches back to IEEE 802.1 spanning tree mode. This indicates that no LAN Bridge 100 spanning tree mode bridges remain in the network.

6.13 Hierarchical Bridge Structures

Figure 6–17 shows an overview of the hierarchical structure of a typical extended LAN. The sample network uses an FDDI ring with concentrators and bridges to interconnect the three buildings. Two of the buildings are approximately 200 meters (656 feet) apart. The third building is approximately 6.4 kilometers (4 miles) away from the other two and is connected to them by a fiber optic link. The FDDI ring may also contain other concentrators and DAS bridges to connect to other LANs in the network.

On each floor of each building, nodes that share a common environment are connected to their own LAN segment. Figure 6–18 and Figure 6–19 show how the LAN segments are interconnected in each building. In buildings 1 and 2, each of the office segments, lab segments, and computer room segments is connected through a two-port bridge to a concentrator for that floor. The bridges keep the local traffic local, thereby reducing the traffic load on individual LAN segments throughout the network. The concentrators for each floor in the building all connect to the FDDI ring through one concentrator dedicated to that building.

Building 3 has both system segments and LAT segments, and uses two multiport bridges (DECbridge 600 series) to isolate the two sets of segments from one another. A concentrator connects the multiport bridges to the 100-Mb/s fiber optic cable from building 1 and 2. As shown in Figure 6–19, a port on one of the two multiport bridges is used as a 10-Mb/s backup link between building 3 and buildings 1 and 2.

Figure 6–17: Extended LAN Example (Overview)

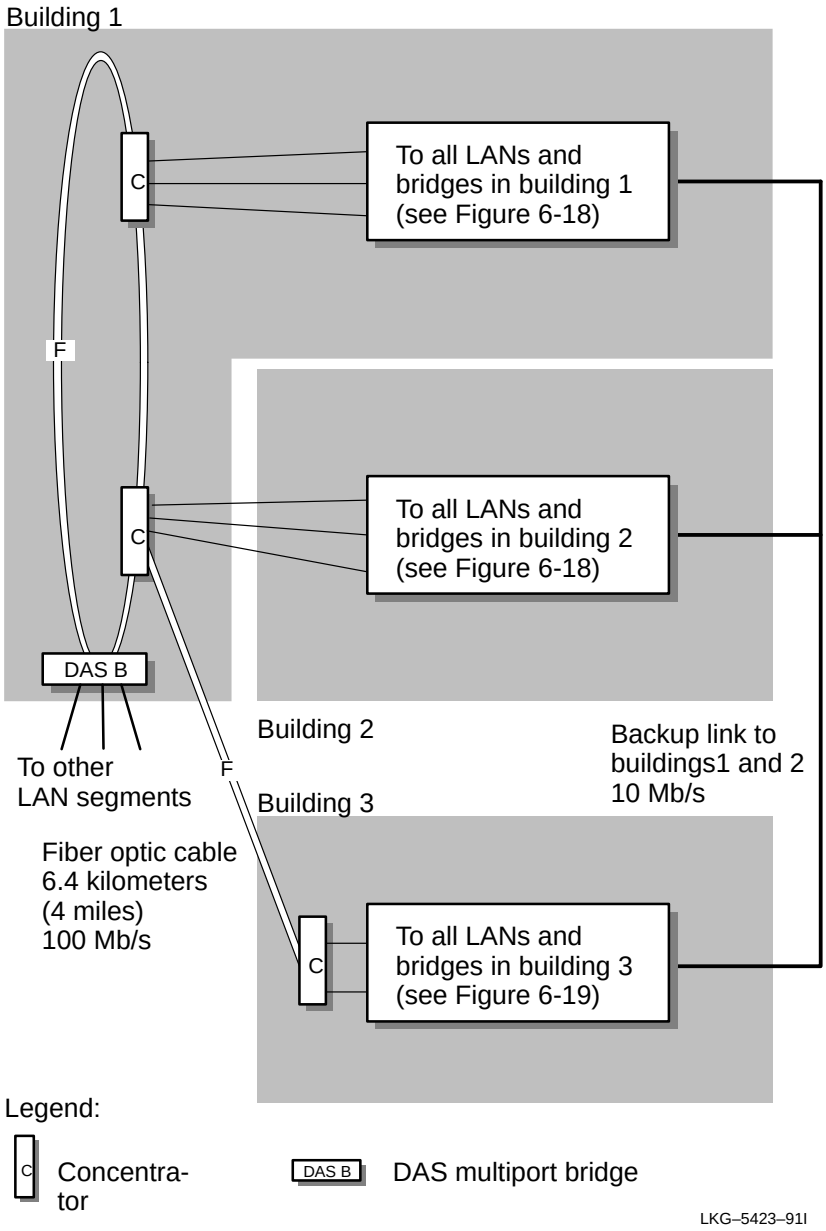


Figure 6–18: Extended LAN Example (Buildings 1 and 2)

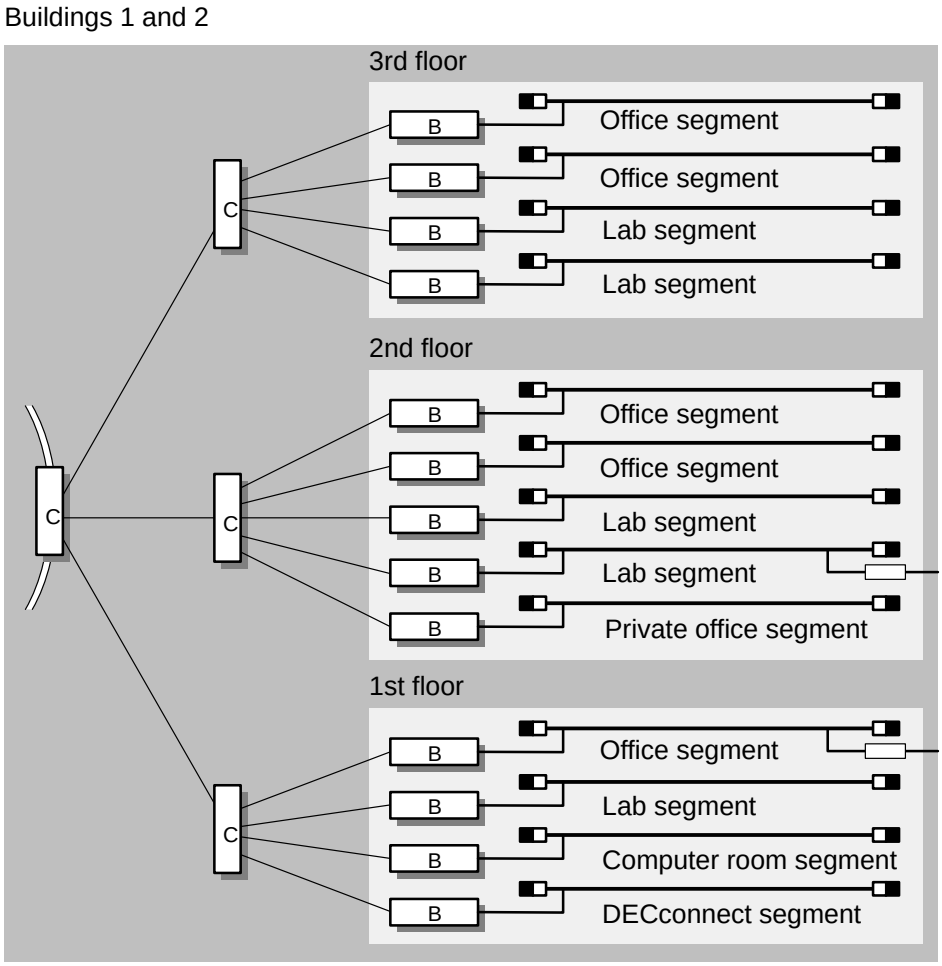
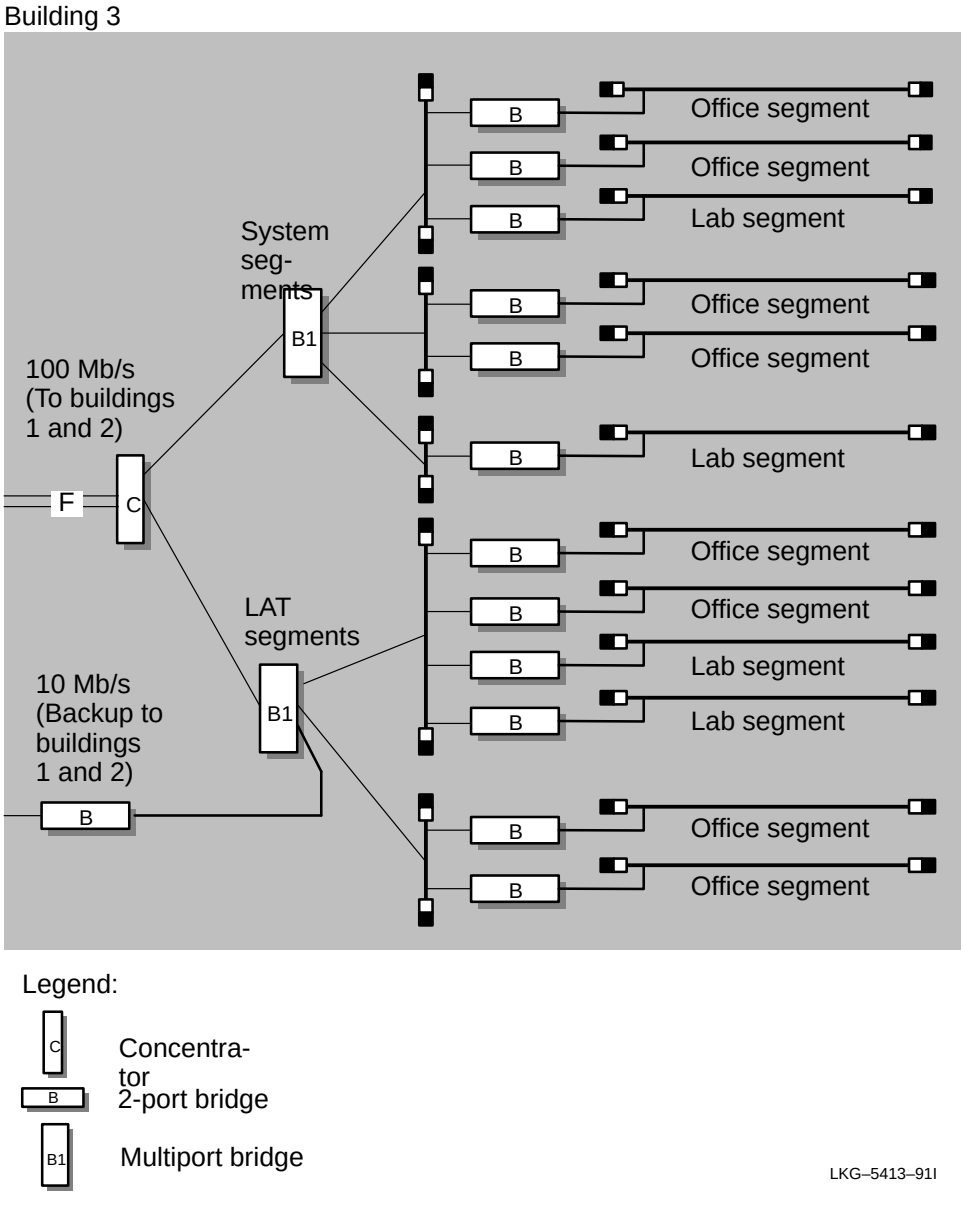


Figure 6–19: Extended LAN Example (Building 3)



7

Network Troubleshooting Methodology

This chapter describes steps that you should complete to develop a firm foundation on which to base operational and diagnostic tests of your network. Effective network troubleshooting requires that you achieve firm baseline knowledge of your network topology before you start the actual troubleshooting.

7.1 Knowing Your Network

During network troubleshooting, it helps to have a good understanding of how your network operates under normal circumstances. This can help you to recognize unusual circumstances faster and allows you to focus on the relevant troubleshooting information. Also, if you maintain a current knowledge of your network, you will be better able to uncover abnormalities in performance or operation that signal potential problems. This section explains what you need to know about your network, including its topology, normal performance, and normal use.

7.1.1 Network Topology

Network topology refers to the physical and logical location of devices in a network.

Physical location refers to where a network device is stationed, for example, in a computer lab, a user's office, a factory floor, an office communications cabinet (OCC), or other location.

Logical location refers to the functional interconnections between devices. For example, a node that is logically adjacent to another node can be physically located on another floor of a building. Functionally, the adjacent node is the closest node in terms of network hops.

Up-to-date maps of the physical and logical locations of all the devices in a network are critical for successful network troubleshooting. Maps can help you understand the extent or overall impact of a failure and can help you isolate problems to a particular LAN, LAN segment, or even a service on a LAN segment.

You can perform simple tests from your office to isolate the source of a problem and then match your findings to the problem device on the network map. An accurate map allows you to go directly to the correct physical source of the problem with the proper tools and begin solving the problem. Without an accurate network map, you can waste valuable time looking for the problem device rather than solving the problem.

The following software tools can help you maintain topological information and network maps:

- DECelms (DEC Extended LAN Management Software)—Allows users logged in to a VAX host to control and monitor any LAN Bridge in an extended LAN. In addition, DECelms provides statistics gathered by the bridge to help monitor and troubleshoot LANs.
- NMCC/VAX ETHERnim—Gathers information about Ethernet nodes, verifies that nodes are reachable, graphically displays the LAN topology, and monitors Ethernet traffic.
- NMCC/DECnet Monitor—Maintains wide area network topology maps.

If these tools are not available, you may want to generate network maps manually. Regardless of the method you use to generate the maps, the maps must be up to date to be of any value. Be sure to monitor any changes to circuits or devices in your network and to update your maps with this changed information regularly.

7.1.2 Network Performance

If you understand the normal range of performance in your network, you can better determine when the network performs poorly due to a network problem. You can use tools such as the LAN Traffic Monitor for LANs and the NMCC/DECnet Monitor for WANs to accumulate information on historical performance and error characteristics. You can also use the NMCC/VAX ETHERnim and the Network Control Program (NCP) to accumulate performance information.

The important concepts to understand concerning network performance are thresholds and usage peaks. A *threshold* is the maximum value set for a parameter. A *usage peak* is the maximum level of user activity that the network can withstand before performance is adversely affected.

When thresholds are reached or when user activity reaches a peak, transient and intermittent performance problems may begin to surface. Performance problems can be the most difficult to isolate, primarily because they are often transient and intermittent. For example, when traffic and queuing requests increase, bandwidth thresholds can be reached and performance problems can occur. The NMCC/DECnet Monitor's graphics display can help uncover problems related to bandwidth and utilization thresholds and can help identify whether insufficient bandwidth is the cause of performance problems.

Maintaining historical performance data is essential for troubleshooting transient and intermittent problems. For example, if you have a transient problem, you can compare the historical performance data against the current performance data and evaluate the differences to help isolate the source of the problem. Historical performance data is also very useful in trend analysis for network growth and network planning.

7.1.3 Network Usage

In understanding the typical use of your network, you need to be aware of the following:

- The applications running on your network
- The times when those applications are running
- The peak performance periods

You should know the types of applications on your network and the typical traffic patterns that the applications produce. For example, some applications can cause performance behavior that is different than normal operation, but is still within accepted bounds. Understanding this helps you determine when a change in performance is acceptable and can be ignored and when it is not acceptable and requires action on your part.

You should also be aware of any unusual performance that a particular application causes, and the possible effects it may have on other applications running at the same time. Understanding the performance of the applications in your network allows you to predict peak usage time, anticipate problems before they occur, and design your network to accommodate the needs of users who require these applications.

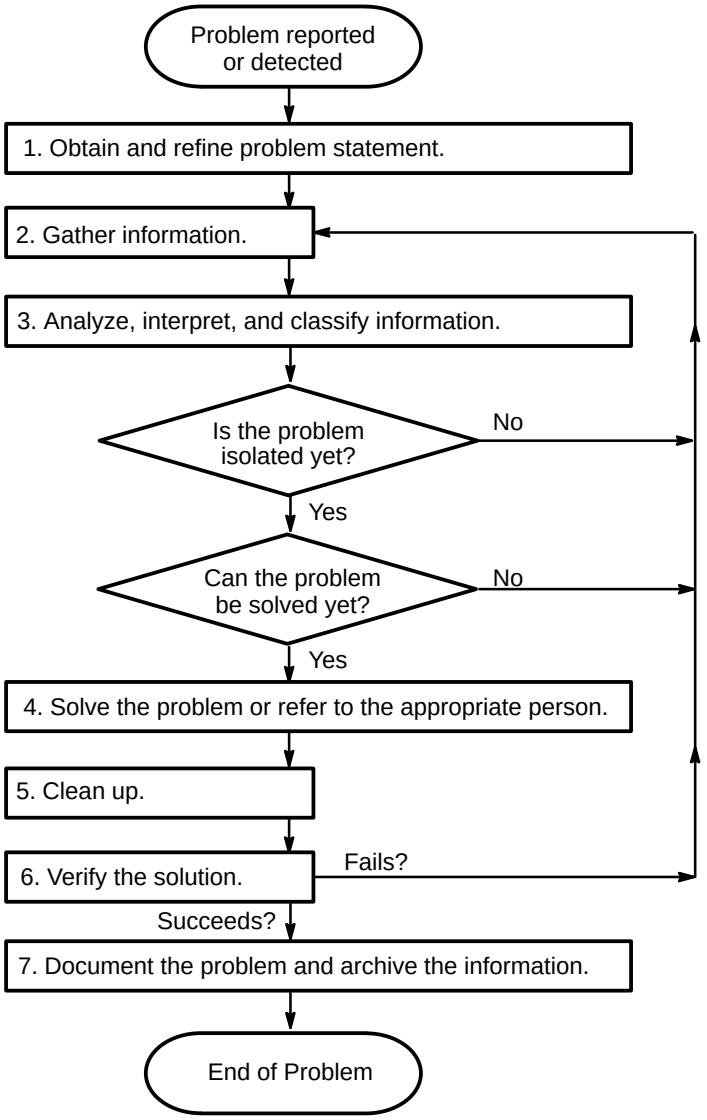
7.2 Overview of the Network Troubleshooting Methodology

This section shows you how to apply the knowledge of your network in a methodical way to resolve any network problem. This network troubleshooting methodology consists of the following steps:

1. Obtain and refine a problem statement.
2. Gather information about the problem.
3. Analyze, interpret, and classify the information.
4. Solve the problem or refer to the responsible person.
5. Clean up any parameters or files created for testing, and remove any test equipment such as loopback connectors.
6. Verify the solution.
7. Document the problem and archive the solution.

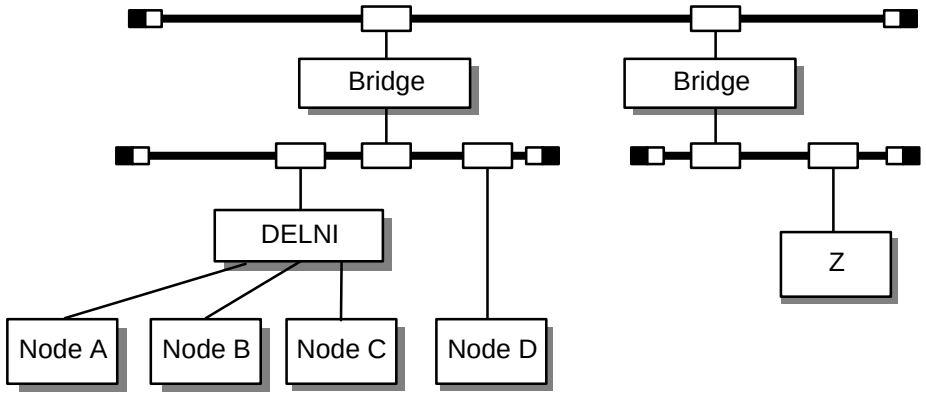
The flowchart in Figure 7–1 shows the relationships between the seven steps of the network troubleshooting methodology. This flowchart is based on the assumption that you understand your network's topology, architectures, software, performance, and normal use, and can apply this knowledge throughout the process. The remaining sections of this chapter describe the troubleshooting methodology in more detail.

Figure 7–1: Network Troubleshooting Methodology



Gathering, analyzing, interpreting, and classifying information are steps that you may need to perform repeatedly throughout the troubleshooting process. The following example shows how these steps apply to solving the problem, "Remote node is not currently reachable." The example uses the network configuration shown in Figure 7–2, where node Z cannot reach node A.

Figure 7–2: Remote Node Unreachable—Example



To solve the problem, complete the following steps:

1. Ensure that node A's address is correctly defined in node Z's node database, and that node Z is on the network.

2. Verify that node Z cannot reach node A by using the following command:

   ```
   $ DIRECTORY A::
   ```

   If this command fails, you receive the error, "Remote node is not currently reachable." Because an unreachable node is a problem in the physical, data link, or routing layers of the Digital Network Architecture (DNA), you can focus on problems in those layers and rule out the user's application as the source of the problem.

   This step helps you to refine the problem statement so that you can focus your troubleshooting efforts appropriately.

3. Assuming the Local Area Transport protocol is active, verify whether the problem is specific to the LAN by trying to connect to the node through LAT.

   If you can reach the node through LAT but not through DECnet, you can focus your efforts on the lower layers of DECnet on node A.

   In this step, you gather information, then analyze, interpret, and classify the information to help determine the cause of the problem, and narrow the focus of your efforts.

4. If you cannot reach the node through LAT, then use an accurate LAN map or the NMCC/VAX ETHERnim display as a guide to the network to try to establish connections to other nodes on the same LAN segment as node A.

   If you cannot reach any other nodes on the segment (nodes B, C, and D), the problem is a LAN segment problem. If you can reach some nodes on the segment (for example, node D) but not others (nodes A, B, and C), the problem may be related to the cable between the DELNI and the transceiver that connects the DELNI to the Ethernet.

   If all connections to the segment fail, the problem is probably related to a bridge or repeater failure, a cable failure, or a screaming node. If only node A fails, the problem is specific to node A.

   In this step, you gather more information, then analyze, interpret, and classify the information to help determine the cause of the problem.

The preceding example illustrates the following points:

- Knowing DNA helps you identify the problem as a physical or data link layer problem.
- Knowing LAN architecture helps you identify the problem as LAN specific.
- Knowing the topology and how to use a network map helps you isolate the problem further.
- Knowing how to use network tools helps you perform tests.
- Testing connectivity to other remote nodes helps you eliminate reachable nodes from the troubleshooting effort and isolate the problem to a LAN segment.

7.3 Analyzing, Interpreting, and Classifying Information

This step of the troubleshooting process involves methodically evaluating the information you collect. Although this step includes several actions, it is considered as one step because the actions are so interrelated. During this step, keep in mind the goal of isolating the problem physically to the node, LAN, or WAN levels, and logically to a specific architecture and layer of that architecture.

In *analyzing* the information, you examine the elements of information and the relationship of each element to the others.

In *interpreting* the information, you evaluate the elements of information and the relationships among the elements to get an understanding of the problem.

In *classifying* the data, you group the information so you can rule out certain problems and begin to isolate the source of the problem.

The remainder of this section discusses classifications of errors by the following categories:

- Extent of the problem
- Types of network errors
- Sources of network errors

As shown in Figure 7–1, you may need to analyze, interpret, and classify information repeatedly as you gather more specific information about the problem you are solving. The continual information gathering throughout the troubleshooting process, followed by analysis, interpretation, and classification of the information, helps you isolate the source of the problem and allows you to eventually solve the problem.

7.3.1 Extent of the Problem

The extent of a network problem refers to the bounds of its disturbance on the network. For example, some problems interfere with correct operation of a single node, while other problems interfere with correct operation of multiple nodes, an entire LAN, or even an entire WAN.

Classifying the extent of the problem continues to focus your efforts, limits the scope of your investigation, and helps you select an appropriate approach to solving the problem. After you classify a problem as specific to a single node, LAN, or WAN, you further define and classify the problem within that area.

The following list defines node, LAN, and WAN problems:

- **Node problems** — A node problem is limited to a specific node, which in this manual is generally assumed to be a VMS node. A node problem generally involves parameters set in AUTHORIZE, NCP, LATCP, and SYSGEN.
- **LAN problems** — A LAN problem is specific to the following:
  - Network protocols, such as LAT, SCA, and MOP, and SMT
  - Network design rules, such as repeater rules, bridge rules, and cabling rules
  - Network hardware, such as bridges, repeaters, connectors, and cabling

  As you begin to isolate the problem source, you may find that a LAN problem is related to a specific node. Nevertheless, it is helpful to begin classifying the extent of the problem as LAN-wide until you have more information to determine otherwise.
- **WAN problems** — A WAN problem extends into a point-to-point Digital Data Communications Message Protocol (DDCMP) environment. WAN problems include: circuit down, router unavailable, partitioned area, and remote node access problems. As with LAN problems, you may find that a WAN problem is related to a specific node. Again, it is helpful to at least begin classifying the extent of the problem as WAN-wide until you have more information to determine otherwise.

7.3.2 Types of Network Errors

This section defines four types of network errors: hard, inconsistent, intermittent, and transient.

- **Hard errors** — Hard errors are consistently reproducible and consistently produce the same error message or symptom when reproduced under the same circumstances.

- **Inconsistent errors** — Inconsistent errors occur when several different error messages or symptoms ultimately have the same underlying cause. These errors generally involve several protocols or layers of an architecture. The different error messages result from the ways different applications encounter the problem in the protocols and the architectural layers. You can usually reproduce an inconsistent error.

  For example, when trying to mail a file to a remote node, a user may get the following error message:

```
%MAIL-E-SENDERR, error sending to user SMITH at NODEID
%MAIL-E-PROTOCOL, network protocol error)
-SYSTEM-F-LINKABORT, network partner aborted logical link
```

  However, if the user tries to copy the file to the remote node, the user receives the following error:

```
%COPY-E-OPENOUT, error opening NODEID::USER$31:[SMITH]MYFILE.TEMP;1 as
    output
-RMS-F-FUL, device full (insufficient space for allocation)
%COPY-W-NOTCOPIED, USER$25:[JONES]MYFILE.TEMP;1 not copied
```

  Both of these problems are related to a lack of disk space for the default DECnet account. However, the errors are different because the means of accessing the remote node are different—in one case through mail, and in the other through FAL.

- **Intermittent errors** — Intermittent errors show up occasionally and always display the same error message or symptoms for the same circumstances. You can occasionally reproduce intermittent errors.

  For example, setting up a cluster alias improperly may result in intermittent errors. An error occurs when an improperly configured node in the cluster is requested to perform an alias node function. The user only receives an error when the request to the cluster goes to the improperly configured node. For example, if a remote user tries to establish a connection to the cluster, the connection may fail if the node receiving the request is the misconfigured node. However, if the user tries the request again, and the node receiving the request is another, properly configured node, the request succeeds.

  Intermittent errors may also result when threshold values for various parameters are reached. These thresholds are usually sufficient for normal use, but during peak use, the thresholds may be reached and errors can result.

- **Transient errors** — Transient errors occur only occasionally and can rarely be reproduced. Because you cannot reliably reproduce transient errors, they are by far the most difficult errors to troubleshoot.

  As with intermittent errors, transient errors may result when threshold values for various parameters are reached. Because transient errors tend to occur at peak usage times, historical performance data is helpful in determining the cause of the problem.

7.3.3 Sources of Errors

This section describes potential sources of network problems, including user, hardware, software, and configuration errors. Understanding the possible sources of errors enables you to ask questions during the troubleshooting process that help to narrow the scope of the problem. This, in turn, allows you to quickly isolate the source of the problem.

- **User errors** — User errors include typing errors, command syntax errors, improper use of hardware or software due to an unclear understanding of its function, and poor applications programming. User errors can result in hard, inconsistent, intermittent, and transient failures.

- **Hardware errors** — Hardware errors include failed devices, loose connections, faulty or noisy circuitry, and lack of power. Most of these errors result in hard errors, although loose connections and noise on the line can cause intermittent problems. Noisy or dirty circuitry may also cause transient errors, only occurring at certain traffic levels or with certain bit patterns.

- **Software errors** — Software errors involve improperly configured software, thresholds being reached, or limitations of the product, such as use of the product in ways other than those intended and specified. Misconfigured software tends to result in hard errors. Thresholds being reached and product limitations tend to result in intermittent and transient errors.

- **Configuration errors** — Configuration errors on a LAN result from failing to conform to recommendations for product installation and use. For example, errors may result from failing to conform to installation and usage guidelines for Ethernet. Performance problems may result from failing to conform to recommendations for logical network configurations. Configuration errors can result in intermittent problems or hard errors.

7.4 Isolating the Source of the Problem

Quick and efficient fault isolation is necessary for resolving network problems. *Fault isolation* is the process used to determine the source of a problem. In many cases, fault isolation is the most difficult and time-consuming part of resolving a network problem. The more orderly and efficient the fault isolation process, the quicker you can return normal network service to your users.

In some cases, isolating the source of the problem may be all that you can do to resolve a problem. For example, you may determine that the problem lies with a common carrier, requires hardware repair, or involves system parameters on a node to which you do not have access. In these cases, you must turn over the responsibility for resolving the problem to the appropriate individual. However, by properly isolating the problem first, you allow other individuals to focus on the repair that needs to be done and help to minimize the overall downtime of network resources.

In isolating the source of a problem, you use the information you accumulated to limit the scope of your investigation from the broadest possible categorization of the problem to the narrowest—eventually focusing on the specific cause of the problem.

This section discusses tools and methods to use for isolating problems to the node, LAN, and WAN levels. Figure 7–3 shows a flowchart of the problem isolation steps.

7.4.1 Isolating Problems to the Node Level

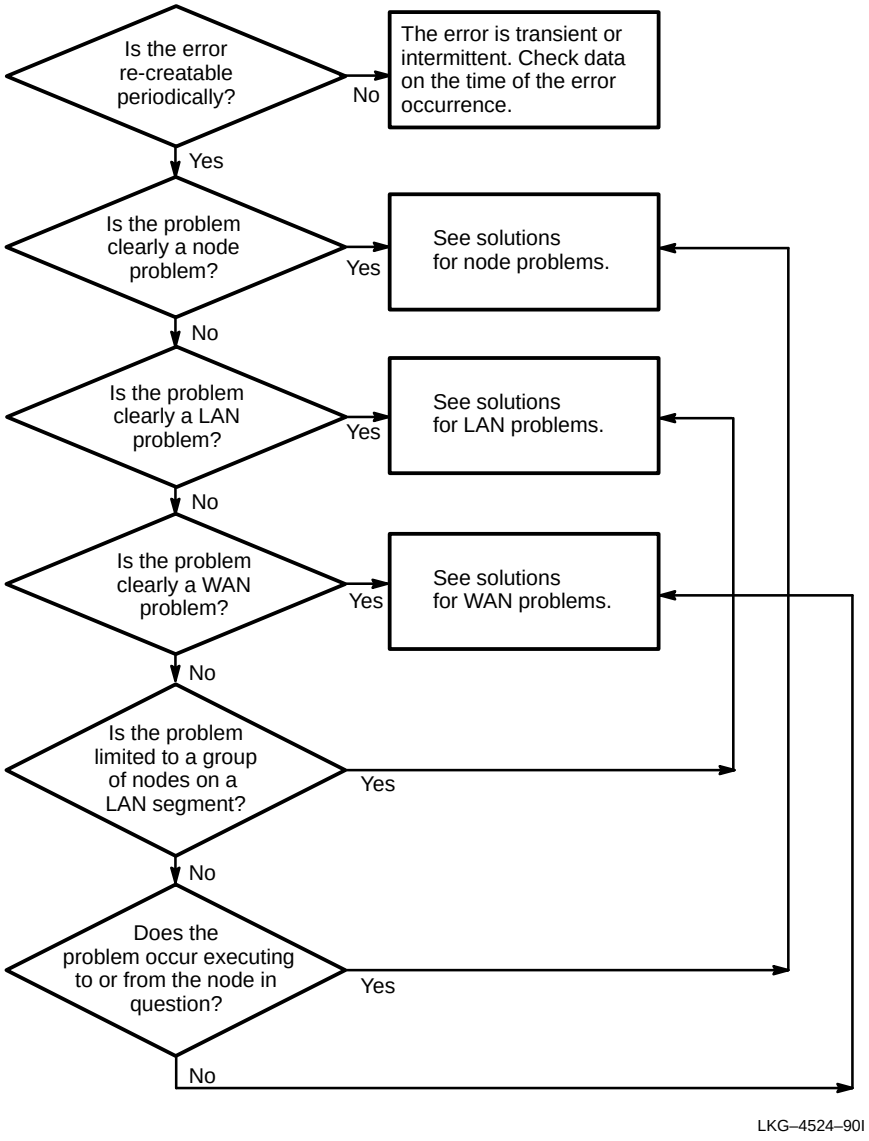
Most problems, including LAN and WAN problems, can eventually be tracked to problems on a specific node. To isolate suspected node problems, retry the failed operation to other nodes to determine whether the problem affects only one node or multiple nodes. For example, if you can get from node A to node C, but not to node B from nodes A or C, then the problem is most likely in node B.

After you isolate a problem to a specific node, you need to determine what exactly on that node is causing the problem. Usually, the problem involves DECnet, Maintenance Operation Protocol (MOP), LAT protocols, or faulty hardware.

When you troubleshoot node problems, keep in mind the protocols and the architectural layers involved in the problem. When you understand which protocol or architectural layer is involved, you automatically narrow down the potential factors contributing to the problem and can focus on the true cause.

Likewise, error messages allow you to refine your understanding of a problem. With practice, you can use error messages to quickly eliminate potential sources of the problem and focus on a specific problem area. For example, once you know that the error message, “Circuit on-starting,” indicates a hardware, cabling, modem, or circuit problem, you can use the VMS error log file to gather more information about the potential hardware problem causing it. Loopback tests provide additional information that can help you isolate the problem to the cabling, modem, or circuit.

Figure 7–3: Isolating the Source of the Problem



The error message, “Device not mounted,” indicates that DECnet has not yet started. Errors that occur while starting DECnet generally result from the network device being unable to accept a DECnet address because another protocol is already using the hardware address. To resolve this problem, you need to stop the other protocols, make sure DECnet starts first, then restart the other protocols.

The error messages, “Login information invalid,” “Network object unknown,” and “Network partner exited,” are all DECnet software errors that occur in the upper layers of DECnet. Knowing this allows you to eliminate potential approaches to the problem and to focus on using the tools most likely to help you isolate the source of the problem for these errors. In resolving these types of problems, you may need to run AUTHORIZE to access the SYSUAF file, or NCP to access the network object database. You may also need to use process creation procedures.

Table 7–1 summarizes the most effective tools for isolating node level problems.

Table 7–1: Tools for Node Problems

Type of Problem	Tools to Use
DECnet and MOP	NCP, AUTHORIZE, SYSGEN, AUTOGEN, ETHERnim, DECelms, LTM, netserver log files
LAT	LATCP, SYSGEN, AUTOGEN, LTM
Hardware	Error log files, loopback connectors

7.4.2 Isolating problems to the LAN Level

Because of the nature of bus architectures such as Ethernet, it can be difficult to isolate a problem device in the event of a failure. However, tools such as NMCC/VAX ETHERnim, combined with a good knowledge of your network topology, can be very helpful in locating the problem source.

After you isolate a problem to a LAN, you can further isolate the problem within the LAN by determining which LAN segment has the problem. When you know the LAN segment involved in the problem, you can focus your efforts on determining which node or nodes on that segment are causing the problem or are being affected by the problem. This is where your knowledge of the topology comes into play. You need to know the segments that make up your LAN, and the individual nodes on each segment.

Most LAN problems result from problems with a single device. For example, a *screaming node* on a LAN (a babbling device) is one node transmitting incorrectly, and a LAN segment down is a bridge or repeater failure or a short-circuited connection.

Other important information to keep in mind for solving LAN problems includes cabling rules, specific protocols, and LAN concepts (CSMA/CD for Ethernet or token rotation time for FDDI, slot time, and round-trip propagation delay). This information helps you understand why a problem occurred and how to prevent future problems from occurring. For example, LAN performance problems may result from configuration errors. Using your knowledge of concepts, cabling, and protocols, you can configure your network properly by locating devices on the LAN for optimal performance.

LAN problems can also involve LAT or downline loading protocols such as MOP, and may be caused by problems in a specific area or with a group of nodes. LAN problems can include segmentation problems, cabling problems, or a terminal server problem.

Table 7–2 summarizes the most effective tools for isolating LAN level problems.

Table 7–2: Tools for LAN Problems

Type of Problem	Tools to Use
LAT	LATCP, TSM, DECserver commands, LTM.
MOP	DECelms, OPCOM, ETHERnim, TSM, NCP, LTM.
Cabling	DECelms, ETHERnim, LTM.
Node	See Table 7–1.

7.4.3 Isolating Problems to the WAN Level

WAN problems only occur in point-to-point environments, and all WAN problems involve circuit, line, or router problems.

The first thing to do when you suspect a WAN problem is to try the failed operation to other nodes on your LAN to eliminate the possibility that the problem is actually LAN-based rather than WAN-based. If you can complete the operation on the LAN, try the operation to other nodes outside the LAN, starting with nodes just outside (adjacent) to the LAN, and working your way out to nodes further and further from the LAN. This helps you determine the extent of the failure.

If you find you cannot complete the operation to other remote nodes, the next step is to trace the routing path to isolate potential causes of the problem. Using NCP to trace the routing path is a manual way of generating network topology information to help with this type of problem. NMCC/DECnet Monitor also helps in quickly pointing out potential WAN problem areas by highlighting failed routers or circuits on its display.

For circuit problems, loopback tests help to determine which components in the path are functional, and which components in the path are not functional.

Router problems can cause unreachable node problems, area partitioning, and circuit state problems. Router problems are mostly node problems, since a router is a node designated to process DECnet traffic from one point to another.

Table 7–3 summarizes the most effective tools for isolating WAN level problems.

Table 7–3: Tools for WAN Problems

Type of Problem	Tools to Use
Circuit	NCP, DECnet Monitor, loopback tests, path traces, LTM.
Line	NCP, DECnet Monitor, loopback tests, path traces, LTM.
Node	See Table 7–1.

7.5 Network Management and Troubleshooting Tools

This section lists the tools available for troubleshooting network problems. For more detailed information, see the product documentation. Table 7–4 summarizes information about the network tools, including the environment in which they are usually used (node, LAN, or WAN) and their primary uses.

Table 7–4: Network Management and Troubleshooting Tools

Tool	Environment	Uses
Authorize Utility	Node	Controlling access to VMS systems, allocating resources to users
LAN Traffic Monitor	LAN	Collecting traffic data for any protocol type on an extended Ethernet
LAT Control Program	Node, LAN	Configuring and controlling LAT protocol on VMS host systems
Network Control Program	Node, LAN, WAN	Configuring and controlling DECnet-VAX networks, monitoring network resources, and testing network resources
NMCC/DECnet Monitor	WAN	Collecting, analyzing, and evaluating network data for Phase III and Phase IV DECnet network nodes, and creating and maintaining databases of node and link information
NMCC/VAX ETHERnim	LAN	Gathering information about Ethernet nodes, verifying that nodes are reachable, displaying network topology, and monitoring Ethernet traffic
DECelms	LAN	Monitoring and controlling LAN bridges
Terminal Server Manager	LAN	Monitoring and controlling terminal servers in an extended LAN

7.6 Solving the Problem

By this point, you should have the problem isolated sufficiently that you can apply a solution. If the required solution is beyond the bounds of your authority or expertise, you may need help from another person.

7.7 Cleaning Up

After you solve a network problem, reset and remove all test facilities that were temporarily installed or enabled for troubleshooting. For example, remove loopback connectors, cancel any requests you made for service from outside vendors, and delete test files. This step ensures that any parameters or facilities that you used to solve the problem do not interfere with the operation of the network.

7.8 Verifying the Solution

The purpose of this step is to test the solution to confirm that the problem is correctly and adequately solved. If the solution does not solve the problem, you need to gather more information and analyze and interpret the data again.

7.9 Documenting the Problem and Solution

Because network problems can be complicated to solve, and because network maintenance is often a team effort, it is helpful to keep a record of the problems that occur on your network and how you solved them. When a problem recurs, you or any other person responsible for maintaining the network can consult the record and move quickly to the solution.

When documenting the problem, be sure to include the symptoms of the problem, when it occurred, the explanation for its occurrence, how you solved it, and the tools you used to solve it.

For more information on troubleshooting network problems, see the *Network Troubleshooting Guide*. It provides detailed troubleshooting procedures for specific network problems.

A

Digital’s Bridge Family

This appendix lists the various bridges available from Digital Equipment Corporation. All of the bridges sold by Digital are IEEE 802.3 and Ethernet compatible. In addition, the DECbridge 500/600 series bridges are also ANSI FDDI compatible. Table A–1 compares the major features for each model.

Table A–1: Comparison of Bridge Features

Features	DECbridge 600 Series	DECbridge 500 Series	LAN Bridge 200	LAN Bridge 150	LAN Bridge 100
Destination address filtering	yes	yes	yes	yes	yes
Source address filtering	yes	yes	yes	no	no
Protocol filtering	yes	yes	yes	no	no
Filter rate (packets/second)	480,000	460,000	29,760	24,272	24,272
Forward rate [1] (packets/second)	22,200	14,880	14,880	13,404	13,404
FDDI bridge [2]	yes	yes	no	no	no
802.3/Ethernet ports	3	1	2	2	2
Integrated ThinWire interface port	no	yes	yes	no	no
Spanning tree [3]	yes	yes	yes	yes	yes
LAN Traffic Monitor	no	no	no	yes	yes
Password protection	yes	yes	yes	yes	no
Upgrade kit [4]	yes	yes	N/A	N/A	yes
Network device upgrade feature [5]	yes	yes	no	no	no
Full duplex	no	no	yes	no	no
Synchronous line	no	no	no	no	no
Maximum interbridge fiber optic distance [6]	N/A	N/A	10.0 km (6.2 mi)	3.0 km (1.9 mi)	3.0 km (1.9 mi)
Microwave support	no	no	no	no	no

1 The maximum forwarding rate of theTransLAN bridge depends on the model. The maximum forwarding rate of theTransLAN III is 2000 packets per second (pps), the TransLAN IV is 3000 pps, and the TransLAN 350 is 5000 pps.

2 All DECbridge 500/600 series bridges (except the DECbridge 600) have either a single attachment station (SAS) or dual attachment station (DAS) connection to FDDI, depending on the model. The FDDI connection(s) can be multimode or single-mode, also depending on the model.

Table A–1 (Cont.): Comparison of Bridge Features

Features	DECbridge 90	METROWAVE Bridge	TransLAN
Destination address filtering	yes	yes	yes
Source address filtering	no	no	yes
Protocol filtering	yes	no	yes
Filter rate (packets/second)	29,694	24,272	14,880
Forward rate [1] (packets/second)	14,847	13,404	2000-5000
FDDI bridge [2]	no	no	no
802.3/Ethernet ports	2	2	2 - 8
Integrated ThinWire interface port	yes	no	no
Spanning tree [3]	yes	yes	yes
LAN Traffic Monitor	no	N/A	no
Password protection	yes	yes	yes
Upgrade kit [4]	N/A	yes	N/A
Network device upgrade feature [5]	yes	no	no
Full duplex	no	no	yes
Synchronous line	no	no	yes
Maximum interbridge fiber optic distance [6]	N/A	N/A	N/A
Microwave support	no	yes	no

3 The LAN Bridge 100 and the TransLAN bridge both implement the LAN Bridge 100 spanning tree algorithm. All other bridges in this table implement either the LAN Bridge 100 spanning tree algorithm or the IEEE 802.1 spanning tree algorithm.

4 The DECbridge 500/600 series has upgrade kits for upgrading between models. The LAN Bridge 100 also has an upgrade kit for upgrading to a LAN Bridge 150. See the *Networks Buyer's Guide* or contact your local sales office for more information.

5 Allows you to upgrade the bridge firmware from a VMS-based or ULTRIX-based host system.

6 Early versions of the LAN Bridge 100 are limited to a maximum of 2.0 km (1.2 mi).

B

Workgroup Bridge

Digital also manufactures a DECbridge 90 workgroup bridge for use in its DEChub 90 Ethernet hub or as a standalone module. DEChub 90 is a multifunction Ethernet backplane that provides mounting, power, and connections for up to eight workgroup LAN products, such as the DECbridge 90.

As shown in Figure B–1, a workgroup consists of a small number of attached devices located relatively close together. The DECbridge 90 is typically used to isolate local workgroup traffic from the Ethernet network backbone.

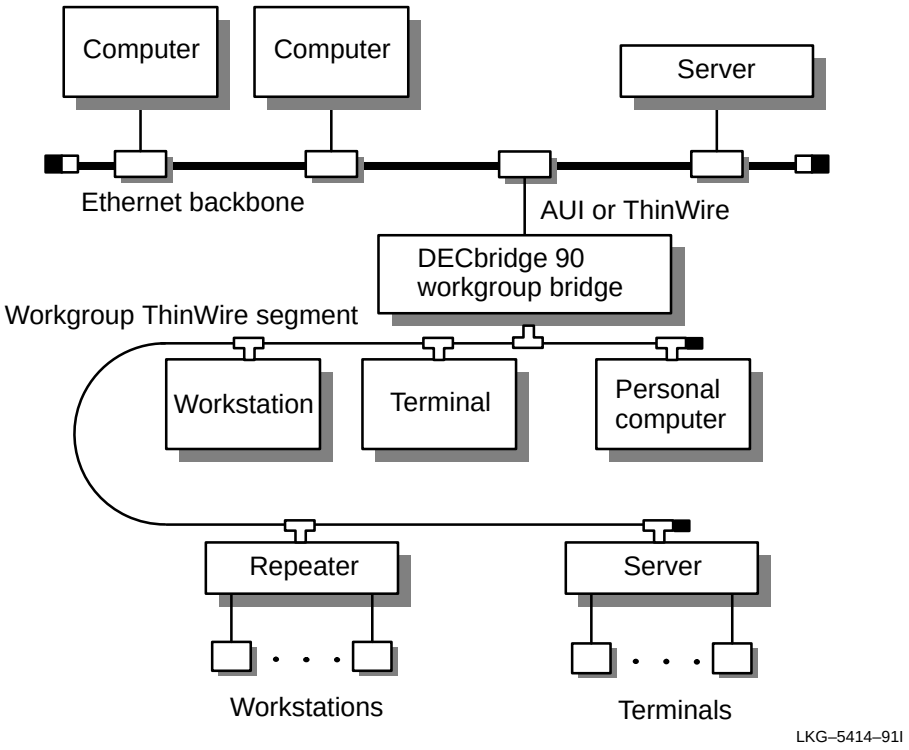
B.1 Features of the DECbridge 90

- Has AUI or ThinWire (switch-selectable) connections to the backbone, and a ThinWire connection to the workgroup.
- Supports a maximum of 200 nodes on the workgroup side.
- Provides protocol filtering.
- Supports the spanning tree algorithm.
- Can be remotely managed by Maintenance Operations Protocol (MOP).
- Has a forwarding rate of 14,847 packets per second and a filtering rate of 29,694 packets per second.

B.2 Major Differences Between the DECbridge 90 and Other Digital Bridges

- The DECbridge 90 cannot do source address filtering.
- The DECbridge 90 has a smaller address database than other Digital bridges.
- The DECbridge 90 learns node addresses only on the workgroup side.
- If the destination address of a packet to the workgroup is unknown, the DECbridge 90 filters the packet. Other Digital bridges would forward the packet.
- The DECbridge 90 must be configured as an end-node bridge, that is, no other bridges can be installed in the workgroup.

Figure B–1: Sample DECbridge 90 Configuration



C

Related Documents

This appendix lists documentation that provides additional information about Digital's bridges and networks. Ordering information is provided at the back of this manual.

C.1 Product Related Documentation

- *Fiber Distributed Data Interface System Level Description* (Order No. EK–DFSLD–SD)

  Describes Digital's Fiber Distributed Data Interface (FDDI) implementation, how it works, and the role of the individual FDDI components. The manual also discusses Digital's approach to network management and the facilities provided by network management software and ring-wide configuration issues.

- *DECelms Use* (Order No. AA–PAK2A–TE)

  Describes how to use DECelms (DEC Extended LAN Management Software) to configure, manage, and monitor the LAN Bridge 100, LAN Bridge 150, LAN Bridge 200, DECbridge 500, and DECconcentrator 500.

- *DECelms Installation* (Order No. AA–PAK1A–TE)

  Explains how to install and verify DECelms (Extended LAN Management Software) on a VMS system.

- *DECelms Reference* (Order No. AA–PBWBA–TE)

  Contains reference information on the DECelms (Extended LAN Management Software) commands.

- *DECmcc Bridge Access Module Use* (Order No. AA–PD1BA–TE)

  Explains how to use the DECmcc (Digital Management Control Center) Bridge Access Module to configure, monitor, and manage bridges.

- *DECbridge 500/600 Series Installation and Upgrade* (Order No. EK–DEFEB–IN)

  Explains how to install the DECbridge 500/600 series units, how to verify their operation once installed, and how to upgrade from one model to another.

- *DECbridge 500/600 Series Problem Solving* (Order No. EK–DEFEB–PS)

  Explains how to troubleshoot and service the DECbridge 500/600 series units, how to remove and replace the field-replaceable units (FRUs). It also includes other technical information about this series of bridges.

- *LAN Bridge 200 Installation* (Order No. EK–DEBAM–IN)

  Explains how to install the LAN Bridge 200 and how to verify its operation. It also describes the LAN Bridge 200 controls and indicators.

- *LAN Bridge 200 Problem Solving* (Order No. EK–DEBAM–PS)

  Provides diagnostics for isolating bridge faults to the field-replaceable unit (FRU). This manual also provides removal and replacement procedures for each FRU.

- *LAN Bridge 150 Installation* (Order No. EK–LB150–IN)

  Explains how to install the LAN Bridge 150 and how to verify its operation. It also describes the LAN Bridge 150 controls and indicators.

- *LAN Bridge 150 Technical Manual* (Order No. EK–LB150–TM)

  Provides a general description of the LAN Bridge 150 unit at the functional component level.

- *LAN Bridge 100 Hardware Installation/Owner's Guide* (Order No. EK–DEBET–UG)

Explains how to install the LAN Bridge 100 and how to verify its operation. It also describes the LAN Bridge 100 controls and indicators.

- *LAN Bridge 100 Technical Manual* (Order No. EK–DEBET–TM)

  Provides a general description of the LAN Bridge 100 unit at the functional component level.

- *Network Troubleshooting Guide* (Order No. EK–339AA–GD)

  Provides an overview of network troubleshooting tools and methodologies, and detailed troubleshooting procedures for specific network problems.

- *LAN Traffic Monitor Installation Guide* (Order No. AA–JP15A–TE)

  Describes the installation of the LAN Traffic Monitor (LTM) software on a VMS system and tells how to downline load the LTM listener software to a LAN Bridge 100 or LAN Bridge 150 hardware unit. This guide also provides installation verification and problem-solving procedures.

- *LAN Traffic Monitor User's Guide* (Order No. AA–JP16A–TE)

  Describes how to use the LAN Traffic Monitor (LTM) feature, the menus, and the informational displays. This guide also provides an overview of the LTM user interface and the LTM listener software.

- *DECconnect System Planning and Configuration Guide* (Order No. EK–DECSY–CG)

  Contains planning requirements and guidelines for configuring DECconnect networks and networks that use DECconnect products. This guide also contains detailed product information for all DECconnect System components.

- D*ECconnect System Facilities Cabling Installation Guide* (Order No. EK–DECSY–FC)

  Provides procedures for properly installing Ethernet coaxial cables, twisted-pair data and voice cables, ThinWire cables, and fiber optic cables within a DECconnect System site. This guide includes installation procedures for devices that are directly related to the facilities cabling (such as transceivers and wallboxes).

C.2 Reference Specifications

The following architecture specifications can be ordered through DECdirect:

- *DECnet–DNA Ethernet Data Link Functional Specification* (Order No. AA–Y298A–TK)
- *DECnet–DNA Ethernet Node Product Architecture Specification V1.0* (Order No. AA–X440A–TK)
- *DECnet–DNA Phase V General Description* (Order No. AA–N149A–TC)
- *DECnet–DNA Maintenance Operation Functional Specification V3.0* (Order No. AA–X436A–TK)
- *DECnet–DNA Network Management Functional Specification V4.0* (Order No. AA–X437A–TK)
- *DECnet–DNA NSP Functional Specification V4.0* (Order No. AA–X439A–TK)
- *DECnet–DNA Routing Layer Functional Specification V2.0* (Order No. AA–X435A–TK)
- *DNA Phase V General Description* (Order No. AA–DNAPV–GD)
- *Ethernet Specification V2.0* (Order No. AA–K759B–TK)
- *DECnet–DNA Data Access Protocol Functional Specification* (Order No. AA–K177A–TK)
- *DECnet–DNA Session Layer Functional Specification* (Order No. AA–K182A–TK)

C.3 Additional Networking Documentation

Additional information about networking products can be found in the following document. Ordering information is provided at the back of this guide.

- *Networks and Communications Product Documentation* (Order No. EK–NACPD–RE)

This guide lists the title and order number for each publication associated with Digital's Telecommunications and Networks products.

For a complete list of the available networking products and for more information, see Digital's *Networks Buyer's Guide*. Customers can receive a catalog by contacting their local sales office.

Glossary

address filtering

The process of preventing frames from being forwarded across a bridge, based on the source or destination address of that frame, or both.

aging time

The spanning tree parameter that controls how long a bridge keeps each learned entry in the forwarding database. If an entry is stored longer than the aging time, the bridge marks that entry as inactive and allows it to be overwritten.

BACKUP (port state)

The state in which a port cannot forward frames but can only monitor the LAN for management and spanning tree information. A port is placed in the BACKUP state to avoid forming a loop in the spanning tree.

bridge

A protocol-independent device that connects local area networks providing logical data link services for all stations on those attached LANs.

bridge address

A sequence of 48 bits that uniquely identifies the bridge. The bridge address is assigned to the bridge during manufacturing.

bridge states

The operational state of a bridge. Digital's bridges have four states: OFF, INIT, OPERATE, and BROKEN.

BROKEN (bridge state)

The state that the bridge enters when it detects a fatal error condition within itself.

BROKEN (port state)

The state in which a port is unable to transmit and receive frames reliably.

DECelms (DEC Extended LAN Management Software)

A VMS layered software product that remotely manages the bridges and wiring concentrators in an extended LAN.

designated bridge

The designated bridge for a LAN is the bridge with the shortest cost path to the root. It connects that LAN with the next LAN closer to the root. Each LAN in an extended network elects one designated bridge.

DISABLED (port state)

A port is disabled by the management entity. In the DISABLED state, a port cannot be used for normal frame forwarding, cannot learn forwarding addresses from received frames, and cannot take part in the spanning tree process. However, in the DISABLED state, a port continues to listen to Hello messages and can receive and transmit management messages under certain conditions.

downline load

The process of sending a software image from a load host to the bridge.

end-to-end delay

The amount of time it takes for a packet to travel from one end of the extended LAN to the other.

entity

A term used to describe functional blocks in the bridge model (for example, management entity and spanning tree entity).

extended LAN

Two or more local area networks (LANs) connected by bridges. The stations connected to these LANs are able to communicate with one another as if they were all on the same LAN.

extended LAN diameter

The number of LANs on the path between the two most distant stations in the extended LAN.

forward delay

A spanning tree parameter that specifies the amount of time a bridge's ports stay in the PREFORWARDING state before entering the FORWARDING state.

FORWARDING (port state)

The state in which the port is fully operational and can forward frames.

forwarding latency

The time required by a bridge to process a frame.

frame lifetime

As it applies to bridges, frame lifetime is the amount of time between when the bridge begins to receive a frame and when it begins to transmit the frame, including the time that the frame waits in the queue to be transmitted on the outgoing data link. As it applies to extended LANs, frame lifetime is the amount of time it takes for a frame to reach its destination on the extended LAN.

frame aging

The process which ensures that the extended LAN does not hold data frames for a longer time interval than the maximum set by certain Transport level protocol timers. The port deletes frames that are held resident too long in the bridge.

Hello interval

A spanning tree parameter that controls how often a bridge sends a Hello message.

Hello messages

Special messages transmitted by all bridges when they are first activated. The information contained in the Hello messages determines which bridges are elected as designated bridges and which bridge becomes the root bridge. When the spanning tree computation process is complete, the root bridge originates the Hello message, which is then propagated down the spanning tree by the other bridges.

INIT (bridge state)

The bridge state entered after the bridge successfully completes its self-test. If no downline loading is to occur, the bridge exits to the PREFORWARDING state.

INIT (port state)

During the INIT state, the associated data link is initializing or self-testing, and all data flow interfaces are off.

Inlink

The port on a bridge that provides the path to the root bridge.

LAN Traffic Monitor (LTM)

An optional mode of operation in Digital's LAN Bridge 100 and LAN Bridge 150, which allows the bridge to act as a *listener* and provide network traffic information.

learning process

The process by which the bridge builds and maintains its address database. By *listening* to network traffic, the bridge notes the source address of each incoming packet and which port received the packet. It then uses that information to update its database.

line cost

A spanning tree parameter set for each line on a bridge and used by the spanning tree algorithm to determine the logical topology of the network. The line cost is between 1 and 255; the default value is 10.

line Identifier

The line identifier assigned by the spanning tree computation process.

LTM listener

A bridge loaded with the LAN Traffic Monitor (LTM) software and dedicated solely to LTM functions. An LTM listener performs no normal bridge functions.

Medium Attachment Unit (MAU)

In a local area network, a device used in a data station to couple the data terminal equipment to the transmission station.

migration bridges

Bridges manufactured by Digital that can dynamically adapt to either the LAN Bridge 100 Spanning Tree Mode or the IEEE 802.1 Spanning Tree Mode, depending on what the network configuration requires. The LAN Bridge 150 and LAN Bridge 200 are migration bridges.

multiport bridges

Bridges that have more than two ports.

NVRAM

Non-volatile random access memory in a bridge. Information in NVRAM is retained even if the bridge loses power.

OPERATE (bridge state)

The normal operational state of a bridge.

path cost

The sum of the line costs along a path between two bridges.

port state

The operational state of a bridge port. Port states are INIT, PREFORWARDING, FORWARDING, BACKUP, DISABLED, and BROKEN.

PREFORWARDING (port state)

During this state, the bridge *learns* new addresses for the address database by monitoring packets received from the LANs. The bridge, however, does not forward packets during the PREFORWARDING state.

protocol filtering

The process of preventing a frame from being forwarded across a bridge, based on the protocol type used by that frame.

root path cost

The sum of the line costs from a bridge to the root bridge.

Root Priority

A spanning tree parameter that determines a bridge's priority for becoming the root of the logical spanning tree. The Root Priority parameter value is used as a prefix to the bridge's address to form the bridge's identification; for example, 128/08–00–2B–2C–08–21.

Station Management Task (SMT)

The entity within a station on the FDDI ring that monitors station activity and exercises overall appropriate control of station activity.

spanning tree

The logical arrangement created by bridges in an extended LAN in which all LANs are connected and there are no loops (that is, there is only one path between any two bridges).

spanning tree algorithm

An algorithm used by bridges to ensure that the extended LAN is configured as a spanning tree.

Index

C

D

E

F

G

H

I

L

M

N

O

P

R

S

T

U

V

W

HOW TO ORDER ADDITIONAL DOCUMENTATION

DIRECT TELEPHONE ORDERS

In Continental USA
and Puerto Rico
call 800–258–1710

In Canada
call 800–267–6146

In New Hampshire
Alaska or Hawaii
call 603–884–6660

DIRECT MAIL ORDERS (U.S. and Puerto Rico*)

DIGITAL EQUIPMENT CORPORATION
P.O. Box CS2008
Nashua, New Hampshire 03061

DIRECT MAIL ORDERS (Canada)

DIGITAL EQUIPMENT OF CANADA LTD.
940 Belfast Road
Ottawa, Ontario, Canada K1G 4C2
Attn: A&SG Business Manager

INTERNATIONAL

DIGITAL
EQUIPMENT CORPORATION
A&SG Business Manager
c/o Digital's local subsidiary
or approved distributor

Internal orders should be placed through Publishing and Circulation Services (P&CS),
Digital Equipment Corporation, 10 Forbes Road, Northboro, Massachusetts 01532–2597

*Any prepaid order from Puerto Rico must be placed
with the Local Digital Subsidiary:
809–754–7575

READER'S COMMENTS

What do you think of this manual? Your comments and suggestions will help us to improve the quality and usefulness of our publications.

Please rate this manual:

	Poor				Excellent
Accuracy	1	2	3	4	5
Readability	1	2	3	4	5
Examples	1	2	3	4	5
Organization	1	2	3	4	5
Completeness	1	2	3	4	5

Did you find errors in this manual? If so, please specify the error(s) and page number(s).

_____

_____

_____

_____

General comments:

_____

_____

_____

_____

Suggestions for improvement:

_____

_____

_____

_____

Name _____ Date _____

Title _____ Department _____

Company _____ Street _____

City _____ State/Country _____ Zip Code _____

DO NOT CUT – FOLD HERE AND TAPE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY LABEL
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

digital™

Telecommunications and Networks Publications
550 King Street
Littleton, MA 01460–1289

DO NOT CUT – FOLD HERE

digital™

Printed in U.S.A.