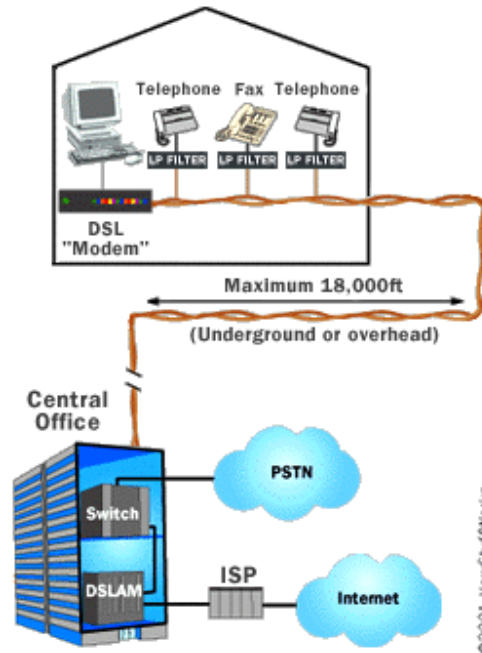


## How DSL Works

by [Curt Franklin](#)

When you connect to the Internet, you might connect through a regular [modem](#), through a [local-area network](#) connection in your office, through a [cable modem](#) or through a **digital subscriber line** (DSL) connection. DSL is a very high-speed connection that uses the same wires as a regular [telephone line](#).



Here are some advantages of DSL:

- You can leave your Internet connection open and still use the phone line for voice calls.
- The speed is much higher than a regular modem (1.5 Mbps vs. 56 Kbps)
- DSL doesn't necessarily require new wiring; it can use the phone line you already have.
- The company that offers DSL will usually provide the modem as part of the installation.

But there are disadvantages:

- A DSL connection works better when you are closer to the provider's central office.
- The connection is faster for receiving data than it is for sending data over the Internet.
- The service is not available everywhere.

In this article, we explain how a DSL connection manages to squeeze more information through a standard phone line -- and lets you make regular telephone calls even when you're online!

## Skinny Voice, Broad Band

If you have read [How Telephones Work](#), then you know that a standard telephone installation in the United States consists of a pair of copper wires that the phone company installs in your home. The copper wires have lots of room for carrying more than your phone conversations -- they are capable of handling a much greater **bandwidth**, or range of frequencies, than that demanded for voice. DSL exploits this "extra capacity" to carry information on the wire without disturbing the line's ability to carry conversations. The entire plan is based on matching particular frequencies to specific tasks.

To understand DSL, you first need to know a couple of things about a normal telephone line -- the kind that telephone professionals call **POTS**, for Plain Old Telephone Service. One of the ways that POTS makes the most of the telephone company's wires and equipment is by limiting the frequencies that the switches, telephones and other equipment will carry. Human voices, speaking in normal conversational tones, can be carried in a frequency range of 0 to 3,400 Hertz (cycles per second -- see [How Telephones Work](#) for a great demonstration of this). This range of frequencies is tiny. For example, compare this to the range of most stereo [speakers](#), which cover from roughly 20 Hertz to 20,000 Hertz. And the wires themselves have the potential to handle

frequencies up to several million Hertz in most cases. The use of such a small portion of the wire's total bandwidth is historical -- remember that the telephone system has been in place, using a pair of copper wires to each home, for about a century. By limiting the frequencies carried over the lines, the telephone system can pack lots of wires into a very small space without worrying about interference between lines. Modern equipment that sends digital rather than analog data can safely use much more of the telephone line's capacity. DSL does just that.

Most homes and small business users are connected to an **asymmetric DSL** (ADSL) line. ADSL divides up the available frequencies in a line on the assumption that most Internet users look at, or download, much more information than they send, or upload. Under this assumption, if the connection speed from the Internet to the user is three to four times faster than the connection from the user back to the Internet, then the user will see the most benefit (most of the time).

## Voice and Data

Precisely how much benefit you see will greatly depend on how far you are from the central office of the company providing the ADSL service. ADSL is a **distance-sensitive technology**: As the connection's length increases, the signal quality decreases and the connection speed goes down. The limit for ADSL service is **18,000 feet** (5,460 meters), though for speed and quality of service reasons many ADSL providers place a lower limit on the distances for the service. At the extremes of the distance limits, ADSL customers may see speeds far below the promised maximums, while customers nearer the central office have faster connections and may see extremely high speeds in the future. ADSL technology can provide maximum downstream (Internet to customer) speeds of up to 8 megabits per second (Mbps) at a distance of about 6,000 feet (1,820 meters), and upstream speeds of up to 640 kilobits per second (Kbps). In practice, the best speeds widely offered today are 1.5 Mbps downstream, with upstream speeds varying between 64 and 640 Kbps.

You might wonder, if distance is a limitation for DSL, why it's not also a limitation for voice telephone calls. The answer lies in small amplifiers called **loading coils** that the telephone company uses to boost voice signals. Unfortunately, these loading coils are incompatible with ADSL signals, so a voice coil in the loop between your telephone and the telephone company's central office will disqualify you from receiving ADSL. Other factors that might disqualify you from receiving ADSL include:

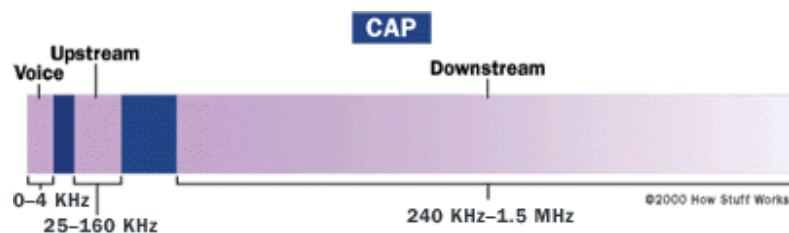
- **Bridge taps** - These are extensions, between you and the central office, that extend service to other customers. While you wouldn't notice these bridge taps in normal phone service, they may take the total length of the circuit beyond the distance limits of the service provider.
- **Fiber-optic cables** - ADSL signals can't pass through the conversion from analog to digital and back to analog that occurs if a portion of your telephone circuit comes through fiber-optic cables.
- **Distance** - Even if you know where your central office is (don't be surprised if you don't -- the telephone companies don't advertise their locations), looking at a map is no indication of the distance a signal must travel between your house and the office.

### Other Types of DSL

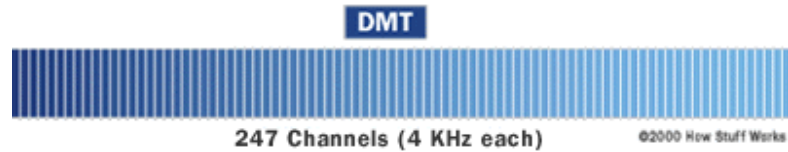
- **Very high bit-rate DSL (VDSL)** - This is a fast connection, but works only over a short distance.
- **Symmetric DSL (SDSL)** - This connection, used mainly by small businesses, doesn't allow you to use the phone at the same time, but the speed of receiving and sending data is the same.
- **Rate-adaptive DSL (RADSL)** - This is a variation of ADSL, but the modem can adjust the speed of the connection depending on the length and quality of the line.

## Splitting the Signal

There are two competing and incompatible standards for ADSL. The official [ANSI](#) standard for ADSL is a system called **discrete multitone**, or DMT. According to equipment manufacturers, most of the ADSL equipment installed today uses DMT. An earlier and more easily implemented standard was the **carrierless amplitude/phase** (CAP) system, which was used on many of the early installations of ADSL.



CAP operates by dividing the signals on the telephone line into three distinct bands: Voice conversations are carried in the 0 to 4 KHz (kilohertz) band, as they are in all POTS circuits. The upstream channel (from the user back to the server) is carried in a band between 25 and 160 KHz. The downstream channel (from the server to the user) begins at 240 KHz and goes up to a point that varies depending on a number of conditions (line length, line noise, number of users in a particular telephone company switch) but has a maximum of about 1.5 MHz (megahertz). This system, with the three channels widely separated, minimizes the possibility of interference between the channels on one line, or between the signals on different lines.



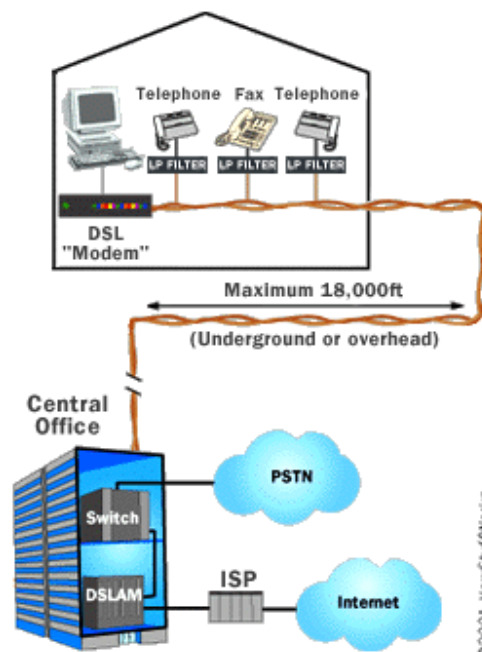
DMT also divides signals into separate channels, but doesn't use two fairly broad channels for upstream and downstream data. Instead, DMT divides the data into 247 separate channels, each 4 KHz wide. One way to think about it is to imagine that the phone company divides your copper line into 247 different 4-KHz lines and then attaches a modem to each one. You get the equivalent of 247 modems connected to your computer at once! Each channel is monitored and, if the quality is too impaired, the signal is shifted to another channel. This system constantly shifts signals between different channels, searching for the best channels for transmission and reception. In addition, some of the lower channels (those starting at about 8 KHz), are used as bidirectional channels, for upstream and downstream information. Monitoring and sorting out the information on the bidirectional channels, and keeping up with the quality of all 247 channels, makes DMT more complex to implement than CAP, but gives it more flexibility on lines of differing quality.



CAP and DMT are similar in one way that you can see as a DSL user. If you have ADSL installed, you were almost certainly given small filters to attach to the outlets that don't provide the signal to your ADSL modem. These filters are **low-pass filters** -- simple filters that block all signals above a certain frequency. Since all voice conversations take place below 4 KHz, the low-pass (LP) filters are built to block everything above 4 KHz, preventing the data signals from interfering with standard telephone calls.

## DSL Equipment

ADSL uses two pieces of equipment, one on the customer end and one at the Internet service provider, telephone company or other provider of DSL services. At the customer's location there is a DSL **transceiver**, which may also provide other services. The DSL service provider has a **DSL Access Multiplexer (DSLAM)** to receive customer connections.



## DSL Transceiver

Most residential customers call their DSL transceiver a "DSL modem." The engineers at the telephone company or ISP call it an **ATU-R**. Regardless of what it's called, it's the point where data from the user's computer or network is connected to the DSL line.



Photo courtesy [Allied Telesyn](#)  
**DSL modem**

The transceiver can connect to a customer's equipment in several ways, though most residential installation uses [USB](#) or 10 base-T [Ethernet](#) connections. While most of the ADSL transceivers sold by ISPs and telephone companies are simply transceivers, the devices used by businesses may combine network [routers](#), network [switches](#) or other networking equipment in the same platform.

## DSLAM

The DSLAM at the access provider is the equipment that really allows DSL to happen. A DSLAM takes connections from many customers and aggregates them onto a single, high-capacity connection to the Internet. DSLAMs are generally flexible and able to support multiple types of DSL in a single central office, and different varieties of protocol and modulation -- both CAP and DMT, for example -- in the same type of DSL. In addition, the DSLAM may provide additional functions including routing or dynamic [IP address](#) assignment for the customers.

The DSLAM provides one of the main differences between user service through ADSL and through [cable modems](#). Because cable-modem users generally share a network loop that runs through a neighborhood, adding users means lowering performance in many instances. ADSL provides a dedicated connection from each user back to the DSLAM, meaning that users won't see a performance decrease as new users are added -- until the total number of users begins to saturate the single, high-speed connection to the Internet. At that point, an upgrade by the service provider can provide additional performance for all the users connected to the DSLAM.

For information on ADSL rates and availability in the United States, go to [Broadband Reports](#). This site can provide information on ADSL service companies in your area, the rates they charge, and customer satisfaction, as well as estimating how far you are from the nearest central office.

For more information on DSL and related topics, check out the links on the next page.

## Lots More Information!

### Related HowStuffWorks Articles

- [How VDSL Works](#)
- [How Home Networking Works](#)
- [How Cable Modems Work](#)
- [How Modems Work](#)
- [How Telephones Work](#)
- [How Fiber Optics Work](#)

### More Great Links

- [WhatIs.com: Fast Guide to DSL](#)
- [xDSL-Related White Papers](#)
- [DSL Forum](#)
- ["DSL for Dummies."](#) by David Angell

#### DSL News, Reviews, Forums

- [Broadband Reports](#)
- [DSL Prime](#)
- [Linksys EtherFast Cable/DSL Router](#) - Review
- [Linksys EtherFast four-port cable/DSL router](#) - Review
- [xDSL.com: Current Headlines](#)

## How Modems Work

by [Marshall Brain](#)

If you are reading this article on your computer at home, it probably arrived via **modem**.

In this edition of [HowStuffWorks](#), we'll show you how a modem brings you Web pages. We'll start with the original 300-baud modems and progress all the way through to the [ADSL](#) configurations!

(Note: If you are unfamiliar with bits, bytes and the ASCII character codes, reading [How Bits and Bytes Work](#) will help make this article much clearer.)

Let's get started with a short recap of how the modem came to be.



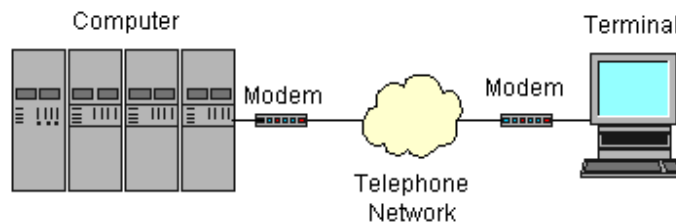
©2002 HowStuffWorks

## The Origin of Modems

The word "modem" is a contraction of the words **modulator-demodulator**. A modem is typically used to send [digital data](#) over a [phone line](#).

The sending modem **modulates** the data into a signal that is compatible with the phone line, and the receiving modem **demodulates** the signal back into digital data. **Wireless modems** convert digital data into [radio signals](#) and back.

Modems came into existence in the 1960s as a way to allow terminals to connect to computers over the phone lines. A typical arrangement is shown below:



In a configuration like this, a **dumb terminal** at an off-site office or store could "dial in" to a large, central computer. The 1960s were the age of **time-shared** computers, so a business would often buy computer time from a time-share facility and connect to it via a 300-bit-per-second (bps) modem.

A dumb terminal is simply a [keyboard](#) and a [screen](#). A very common dumb terminal at the time was called the **DEC VT-100**, and it became a standard of the day (now memorialized in terminal emulators worldwide). The VT-100 could display 25 lines of 80 characters each. When the user typed a character on the terminal, the modem sent the [ASCII code](#) for the character to the computer. The computer then sent the character back to the computer so it would appear on the screen.

When [personal computers](#) started appearing in the late 1970s, **bulletin board systems** (BBS) became the rage. A person would set up a computer with a modem or two and some BBS software, and other people would dial in to connect to the [bulletin board](#). The users would run **terminal emulators** on their computers to emulate a dumb terminal.

People got along at 300 bps for quite a while. The reason this speed was tolerable was because 300 bps represents about 30 characters per second, which is a lot more characters per second than a person can type or read. Once people started transferring large programs and images to and from bulletin board systems, however, 300 bps became intolerable. Modem speeds went through a series of steps at approximately two-year intervals:

- 300 bps - 1960s through 1983 or so
- 1200 bps - Gained popularity in 1984 and 1985
- 2400 bps
- 9600 bps - First appeared in late 1990 and early 1991
- 19.2 kilobits per second (Kbps)
- 28.8 Kbps
- 33.6 Kbps
- 56 Kbps - Became the standard in 1998
- ADSL, with theoretical maximum of up to 8 megabits per second (Mbps) - Gained popularity in 1999

(Check out [How DSL Works](#) and [How Cable Modems Work](#) for more information on the progression of modem technology and current speeds.)

## 300-bps Modems

We'll use 300-bps modems as a starting point because they are extremely easy to understand. A 300-bps modem is a device that uses **frequency shift keying** (FSK) to transmit digital information over a telephone line. In frequency shift keying, a different tone (frequency) is used for the different bits (see [How Guitars Work](#) for a discussion of tones and frequencies).

When a terminal's modem dials a computer's modem, the terminal's modem is called the **originate** modem. It transmits a 1,070-hertz tone for a 0 and a 1,270-hertz tone for a 1. The computer's modem is called the **answer** modem, and it transmits a 2,025-hertz tone for a 0 and a 2,225-hertz tone for a 1. Because the originate and answer modems transmit different tones, they can use the line simultaneously. This is known as **full-duplex** operation. Modems that can transmit in only one direction at a time are known as **half-duplex** modems, and they are rare.

Let's say that two 300-bps modems are connected, and the user at the terminal types the letter "a." The ASCII code for this letter is 97 decimal or 0110001 binary (see [How Bits and Bytes Work](#) for details on binary). A device inside the terminal called a UART (universal asynchronous receiver/transmitter) converts the byte into its bits and sends them out one at a time through the terminal's **RS-232 port** (also known as a [serial port](#)). The terminal's modem is connected to the RS-232 port, so it receives the bits one at a time and its job is to send them over the phone line.

## Faster Modems

In order to create faster modems, modem designers had to use techniques far more sophisticated than frequency-shift keying. First they moved to **phase-shift keying** (PSK), and then **quadrature amplitude modulation** (QAM). These techniques allow an incredible amount of information to be crammed into the 3,000 hertz of bandwidth available on a normal voice-grade phone line. 56K modems, which actually connect at something like 48 Kbps on anything but absolutely perfect lines, are about the limit of these techniques (see the links at the end of this article for more information).

Here's a look inside a typical 56K modem:



All of these high-speed modems incorporate a concept of **gradual degradation**, meaning they can test the phone line and fall back to slower speeds if the line cannot handle the modem's fastest speed.

The next step in the evolution of the modem was **asymmetric digital subscriber line (ADSL)** modems. The word *asymmetric* is used because these modems send data faster in one direction than they do in another. An ADSL modem takes advantage of the fact that any normal home, apartment or office has a **dedicated copper wire** running between it and phone company's nearest mux or central office. This dedicated copper wire can carry far more data than the 3,000-hertz signal needed for your phone's voice channel. If both the phone company's central office and your house are equipped with an ADSL modem on your line, then the section of copper wire between your house and the phone company can act as a purely digital high-speed transmission channel. The capacity is something like 1 million bits per second (Mbps) between the home and the phone company (*upstream*) and 8 Mbps between the phone company and the home (*downstream*) under ideal conditions. The same line can transmit both a phone conversation *and* the digital data.

The approach an ADSL modem takes is very simple in principle. The phone line's bandwidth between 24,000 hertz and 1,100,000 hertz is divided into 4,000-hertz bands, and a **virtual modem** is assigned to each band. Each of these 249 virtual modems tests its band and does the best it can with the slice of bandwidth it is allocated. The aggregate of the 249 virtual modems is the total speed of the pipe.

(For information on the latest DSL technology, see [How DSL Works](#).)

## Point-to-Point Protocol

Today, no one uses dumb terminals or terminal emulators to connect to an individual computer. Instead, we use our modems to connect to an **Internet service provider (ISP)**, and the ISP connects us into the Internet. The Internet lets us connect to any machine in the world (see [How Web Servers and the Internet Work](#) for details). Because of the relationship between your computer, the ISP and the Internet, it is no longer appropriate to send individual characters. Instead, your modem is routing TCP/IP packets between you and your ISP.

The standard technique for [routing](#) these packets through your modem is called the **Point-to-Point Protocol (PPP)**. The basic idea is simple -- your computer's TCP/IP stack forms its TCP/IP datagrams normally, but then the datagrams are handed to the modem for transmission. The ISP receives each datagram and routes it appropriately onto the Internet. The same process occurs to get data from the ISP to your computer. See [this page](#) for additional information on PPP.

If you want to know more about modems, protocols, and especially if you wish to delve into things like PSK and QAM in more detail, check out the links on the next page!

## Lots More Information!

### Related HowStuffWorks Articles

- [How DSL Works](#)
- [How VDSL Works](#)

- [How Cable Modems Work](#)
- [How Web Servers and the Internet Work](#)
- [How Internet Infrastructure Works](#)
- [How Routers Work](#)
- [How Wireless Internet Works](#)
- [How Telephones Work](#)
- [How Fiber Optics Work](#)
- [How Serial Ports Work](#)

## More Great Links!

- [Asynchronous Serial Transmission](#) - good introduction
- [A Brief Introduction to Modem Technology](#)
- [56K Modem FAQ](#)
- [32 V.34 Modems Answer The Call](#)
- [Data communication over the telephone network](#)
- [Modern Quadrature Amplitude Modulation: Principles and Applications for Fixed and Wireless Communications](#), by William Webb, Lajos Hanzo
- [Point-to-Point Protocol](#)
- [The Packet Radio Users Notebook](#)
- [PSK and QAM systems](#)

## How Cable Modems Work

by [Curt Franklin](#)

For millions of people, [television](#) brings news, entertainment and educational programs into their homes. Many people get their TV signal from [cable television](#) (CATV) because cable TV provides a clearer picture and more channels. See [How Cable TV Works](#) for details.

Many people who have cable TV can now get a high-speed connection to the Internet from their cable provider. Cable modems compete with technologies like [asymmetrical digital subscriber lines](#) (ADSL). If you have ever wondered what the differences between DSL and cable modems are, or if you have ever wondered how a computer network can share a cable with dozens of television channels, then read on. In this article, we'll look at how a cable modem works and see how 100 cable television channels and any Web site out there can flow over a single coaxial cable into your home.

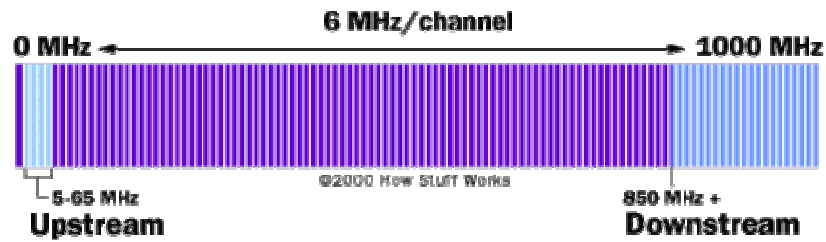
## Extra Space

You might think that a television channel would take up quite a bit of electrical "space," or **bandwidth**, on a cable. In reality, each television signal is given a 6-megahertz (MHz, millions of cycles per second) channel on the cable. The **coaxial cable** used to carry cable television can carry hundreds of megahertz of signals -- all the channels you could want to watch and more. (For more information, see [How Television Works](#).)

In a cable TV system, signals from the various channels are each given a 6-MHz slice of the cable's available bandwidth and then sent down the cable to your house. In some systems, coaxial cable is the only medium used for distributing signals. In other systems, [fiber-optic cable](#) goes from the cable company to different neighborhoods or areas. Then the fiber is terminated and the signals move onto coaxial cable for distribution to individual houses.



Photo courtesy [Motorola, Inc.](#)  
**Motorola SB5100E SURFboard  
 Cable Modem**



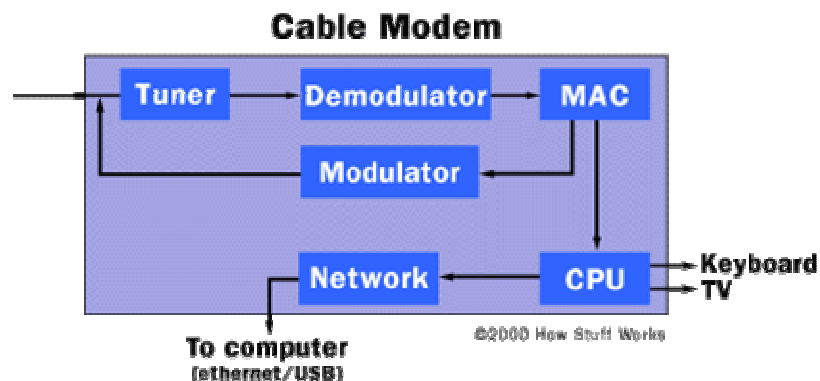
When a cable company offers Internet access over the cable, Internet information can use the same cables because the cable modem system puts **downstream** data -- data sent from the Internet to an individual computer -- into a 6-MHz channel. On the cable, the data looks just like a TV channel. So Internet downstream data takes up the same amount of cable space as any single channel of programming. **Upstream** data -- information sent from an individual back to the Internet -- requires even less of the cable's bandwidth, just 2 MHz, since the assumption is that most people download far more information than they upload.

Putting both upstream and downstream data on the cable television system requires two types of equipment: a **cable modem** on the customer end and a **cable modem termination system (CMTS)** at the cable provider's end. Between these two types of equipment, all the computer networking, security and management of Internet access over cable television is put into place.

## Inside the Cable Modem

Cable modems can be either internal or external to the [computer](#). In some cases, the cable modem can be part of a set-top cable box, requiring that only a [keyboard](#) and [mouse](#) be added for Internet access. In fact, if your cable system has upgraded to digital cable, the new set-top box the cable company provides will be capable of connecting to the Internet, whether or not you receive Internet access through your CATV connection. Regardless of their outward appearance, all cable modems contain certain key components:

- A **tuner**
- A **demodulator**
- A **modulator**
- A **media access control (MAC)** device
- A **microprocessor**



## Tuner

The tuner connects to the cable outlet, sometimes with the addition of a **splitter** that separates the Internet data channel from normal CATV programming. Since the Internet data comes through an otherwise unused cable channel, the tuner simply receives the modulated digital signal and passes it to the demodulator.

In some cases, the tuner will contain a **diplexer**, which allows the tuner to make use of one set of frequencies (generally between 42 and 850 MHz) for downstream traffic, and another set of frequencies (between 5 and 42 MHz) for the upstream data. Other systems, most often those with more limited capacity for channels, will use the cable modem tuner for downstream data and a [dial-up telephone modem](#) for upstream traffic. In either case, after the tuner receives a signal, it is passed to the demodulator.

## Demodulator

The most common demodulators have four functions. A quadrature amplitude modulation (QAM) demodulator takes a [radio-frequency](#)

signal that has had information encoded in it by varying both the amplitude and phase of the wave, and turns it into a simple signal that can be processed by the analog-to-digital (A/D) converter. The A/D converter takes the signal, which varies in voltage, and turns it into a series of digital 1s and 0s. An error correction module then checks the received information against a known standard, so that problems in transmission can be found and fixed. In most cases, the network **frames**, or groups of data, are in [MPEG format](#), so an MPEG synchronizer is used to make sure the data groups stay in line and in order.

## Modulator

In cable modems that use the cable system for upstream traffic, a modulator is used to convert the digital computer network data into radio-frequency signals for transmission. This component is sometimes called a **burst modulator**, because of the irregular nature of most traffic between a user and the Internet, and consists of three parts:

- A section to insert information used for error correction on the receiving end
- A QAM modulator
- A digital-to-analog (D/A) converter

## Media Access Control (MAC)

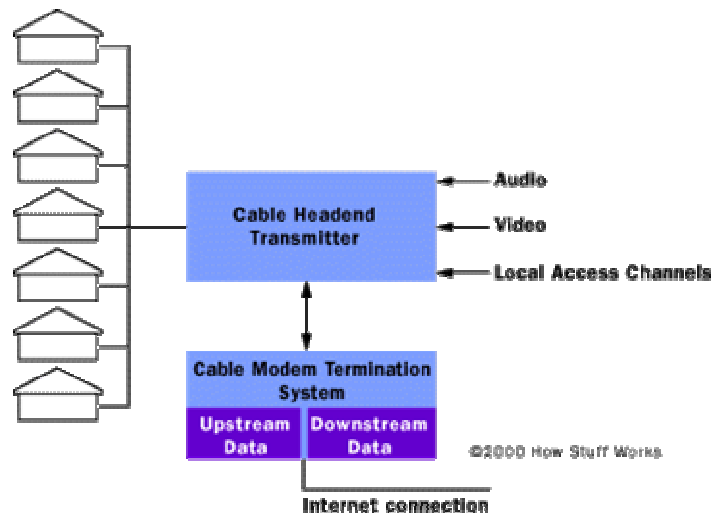
The MAC sits between the upstream and downstream portions of the cable modem, and acts as the interface between the hardware and software portions of the various network **protocols**. All computer network devices have MACs, but in the case of a cable modem the tasks are more complex than those of a normal network interface card. For this reason, in most cases, some of the MAC functions will be assigned to a central processing unit (CPU) -- either the CPU in the cable modem or the CPU of the user's system.

## Microprocessor

The microprocessor's job depends somewhat on whether the cable modem is designed to be part of a larger computer system or to provide Internet access with no additional computer support. In situations calling for an attached computer, the internal [microprocessor](#) still picks up much of the MAC function from the dedicated MAC module. In systems where the cable modem is the sole unit required for Internet access, the microprocessor picks up MAC slack and much more. In either case, [Motorola's PowerPC processor](#) is one of the common choices for system designers.

## Cable Modem Termination System

At the cable provider's head-end, the CMTS provides many of the same functions provided by the DSLAM in a [DSL](#) system. The CMTS takes the traffic coming in from a group of customers on a single channel and routes it to an Internet service provider (ISP) for connection to the Internet. At the head-end, the cable providers will have, or lease space for a third-party ISP to have, [servers](#) for accounting and logging, [Dynamic Host Configuration Protocol](#) (DHCP) for assigning and administering the [IP addresses](#) of all the cable system's users, and control servers for a protocol called CableLabs Certified Cable Modems -- formerly [Data Over Cable Service Interface Specifications](#) (DOCSIS), the major standard used by U.S. cable systems in providing Internet access to users.



The downstream information flows to all connected users, just like in an [Ethernet](#) network -- it's up to the individual network connection to decide whether a particular block of data is intended for it or not. On the upstream side, information is sent from the user to the CMTS -- other users don't see that data at all. The narrower upstream bandwidth is divided into slices of time, measured in milliseconds, in which users can transmit one "burst" at a time to the Internet. The division by time works well for the very short commands, queries and addresses that form the bulk of most users' traffic back to the Internet.

A CMTS will enable as many as 1,000 users to connect to the Internet through a single 6-MHz channel. Since a single channel is capable of 30 to 40 megabits per second (Mbps) of total throughput, this means that users may see far better performance than is available with standard [dial-up modems](#). The single channel aspect, though, can also lead to one of the issues some users experience with cable modems.

If you are one of the first users to connect to the Internet through a particular cable channel, then you may have nearly the entire bandwidth of the channel available for your use. As new users, especially heavy-access users, are connected to the channel, you will have to share that bandwidth, and may see your performance degrade as a result. It is possible that, in times of heavy usage with many connected users, performance will be far below the theoretical maximums. The good news is that this particular performance issue can be resolved by the cable company adding a new channel and splitting the base of users.

Another benefit of the cable modem for Internet access is that, unlike [ADSL](#), its performance doesn't depend on distance from the central cable office. A digital CATV system is designed to provide digital signals at a particular quality to customer households. On the upstream side, the burst modulator in cable modems is programmed with the distance from the head-end, and provides the proper signal strength for accurate transmission.

For more information on cable modems and related topics, check out the links on the next page.

## Lots More Information

### Related HowStuffWorks Articles

- [How Modems Work](#)
- [How DSL Works](#)
- [How VDSL Works](#)
- [How Cable Television Works](#)
- [How Home Networking Works](#)
- [How Ethernet Works](#)
- [How Wireless Internet Works](#)
- [How Routers Work](#)
- [Which is better to use for a cable modem -- a USB connection or an Ethernet card?](#)
- [Why the difference in speed with my cable modem?](#)
- [How does a T1 line work?](#)

### More Great Links

- [Cable Modem 101](#)
- [What is a Cable Modem?](#)
- [Navas Cable Modem/DSL Tuning Guide](#)
- [Cable-Modem.net](#)
- [Internet Speed Test](#)
- [Cable/DSL Speed Patches](#)
- [CableModemInfo.com](#)
- [CableModemHelp](#)
- [Cable Modem Info Center](#)
- [What's Inside a Cable Modem?](#)
- [CableSense](#) - information on cable modems

### News & Reviews

- [Cable Datacom News](#)
- [CableWorld Magazine](#)

## How LAN Switches Work

by [Jeff Tyson](#)

If you have read other HowStuffWorks articles on [networking](#) or the [Internet](#), then you know that a typical network consists of nodes (computers), a connecting medium (wired or wireless) and specialized network equipment like [routers](#) or hubs. In the case of the Internet, all of these pieces work together to allow your computer to send information to another computer that could be on the other side of the world!

**Switches** are a fundamental part of most networks. They make it possible for several users to send information over a network at the same time without slowing each other down. Just like routers allow different networks to communicate with each other, switches allow different **nodes** (a network connection point, typically a computer) of a network to communicate directly with one another in a smooth and efficient manner.



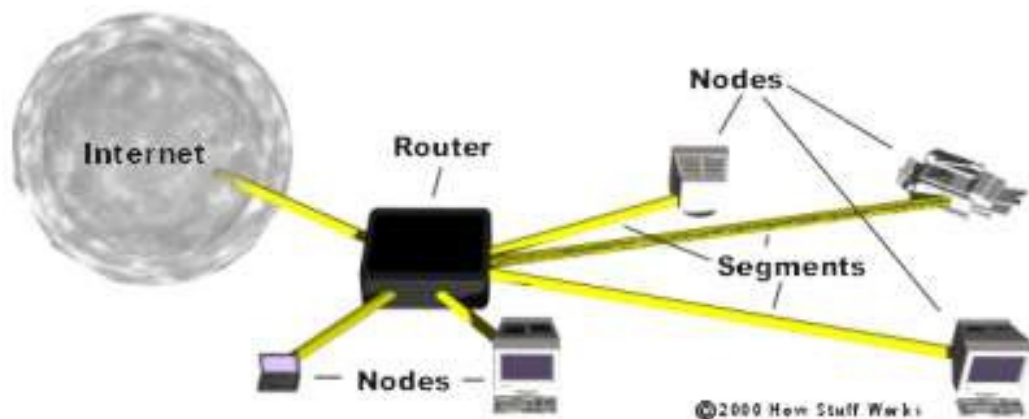
Image courtesy Cisco Systems, Inc.

**Illustration of a Cisco Catalyst switch**

There are a lot of different types of switches and networks. Switches that provide a separate connection for each node in a company's internal network are called **LAN switches**. Essentially, a LAN switch creates a series of instant networks that contain only the two devices communicating with each other at that particular moment. In this edition of [HowStuffWorks](#), we will focus on [Ethernet](#) networks that use LAN switches. You will learn what a LAN switch is and how transparent bridging works, as well as about VLANs, trunking and spanning trees.

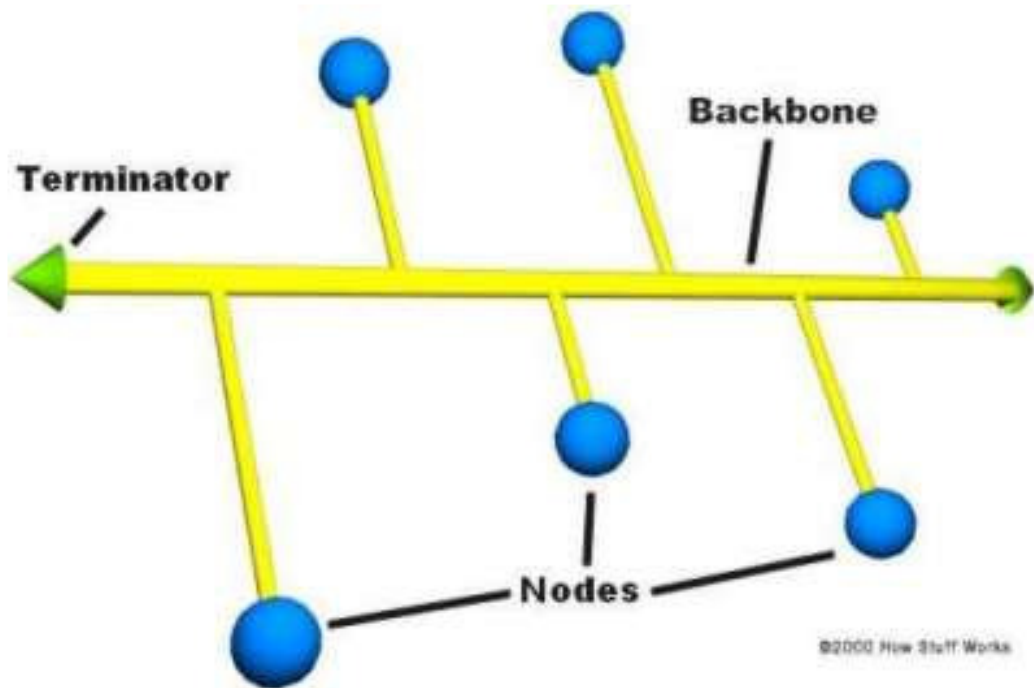
## Networking Basics

Here are some of the fundamental parts of a network:



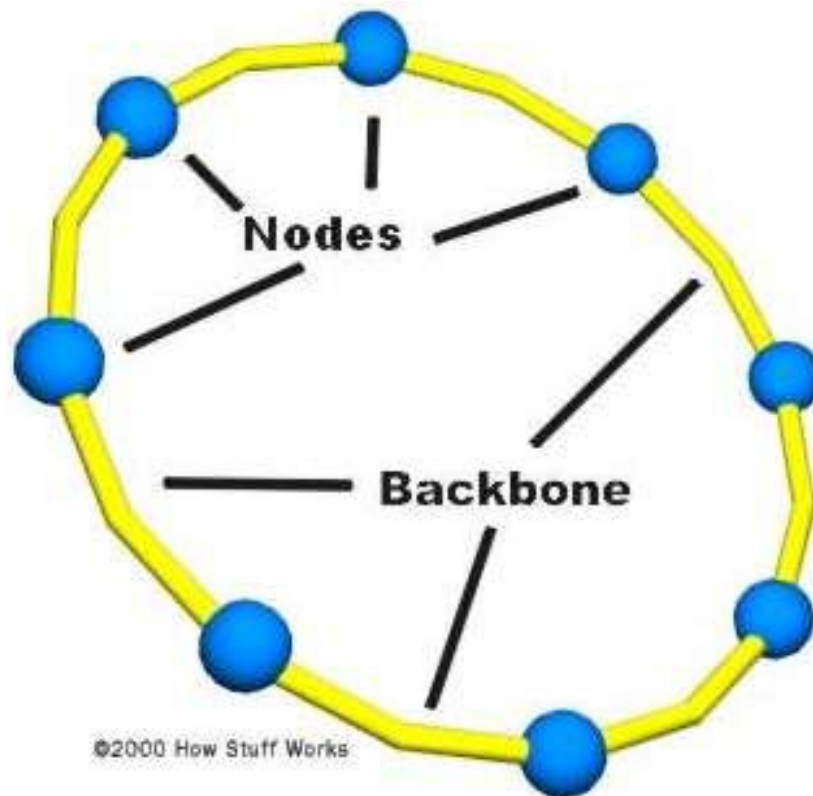
- **Network** - A network is a group of [computers](#) connected together in a way that allows information to be exchanged between the computers.
- **Node** - A node is anything that is connected to the network. While a node is typically a computer, it can also be something like a [printer](#) or [CD-ROM](#) tower.
- **Segment** - A segment is any portion of a network that is separated, by a switch, bridge or router, from other parts of the network.
- **Backbone** - The backbone is the main cabling of a network that all of the segments connect to. Typically, the backbone is capable of carrying more information than the individual segments. For example, each segment may have a transfer rate of 10 Mbps ([megabits](#) per second), while the backbone may operate at 100 Mbps.
- **Topology** - Topology is the way that each node is physically connected to the network. Common topologies include:
  - **Bus** - Each node is **daisy-chained** (connected one right after the other) along the same backbone, similar to [Christmas lights](#). Information sent from a node travels along the backbone until it reaches its destination node. Each end of a bus network must be

**terminated** with a resistor to keep the signal that is sent by a node across the network from bouncing back when it reaches the end of the cable.



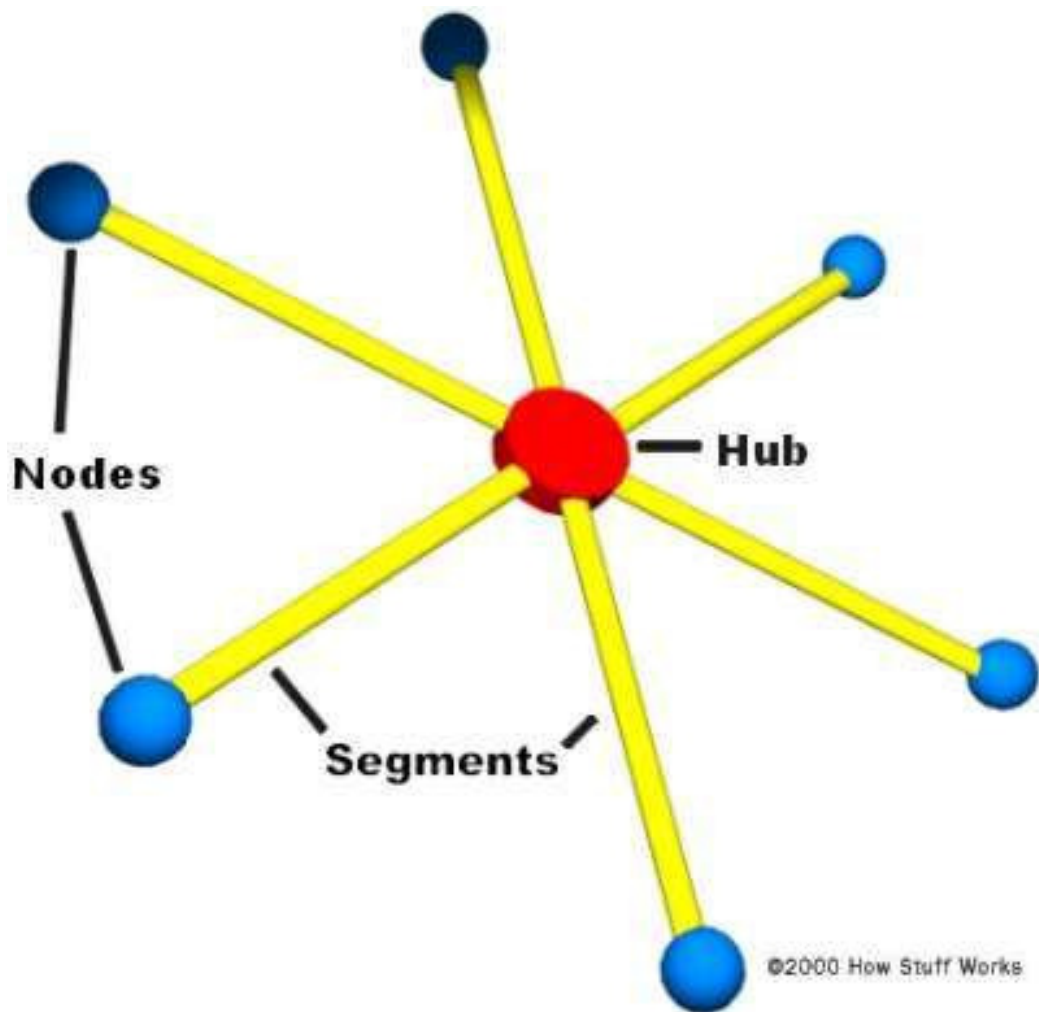
**Bus network topology**

- **Ring** - Like a bus network, rings have the nodes daisy-chained. The difference is that the end of the network comes back around to the first node, creating a complete circuit. In a ring network, each node takes a turn sending and receiving information through the use of a **token**. The token, along with any data, is sent from the first node to the second node, which extracts the data addressed to it and adds any data it wishes to send. Then, the second node passes the token and data to the third node, and so on until it comes back around to the first node again. Only the node with the token is allowed to send data. All other nodes must wait for the token to come to them.



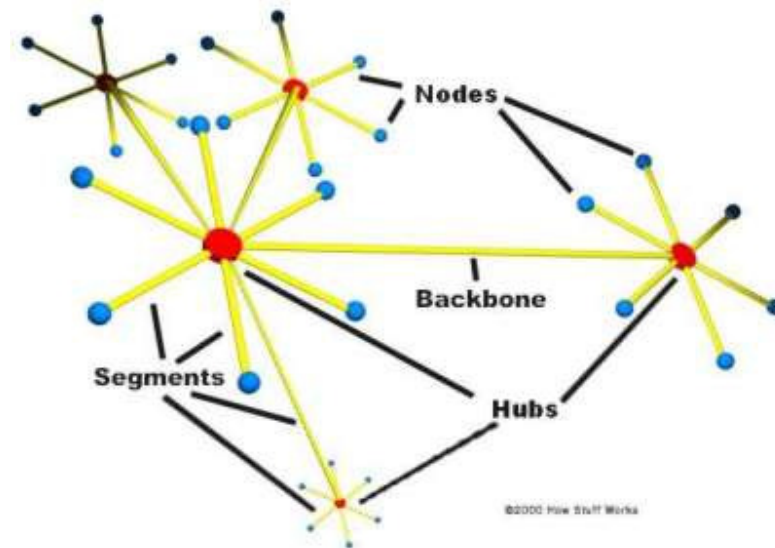
**Ring network topology**

- **Star** - In a star network, each node is connected to a central device called a **hub**. The hub takes a signal that comes from any node and passes it along to all the other nodes in the network. A hub does not perform any type of filtering or routing of the data. It is simply a junction that joins all the different nodes together.



**Star network topology**

- **Star bus** - Probably the most common network topology in use today, star bus combines elements of the star and bus topologies to create a versatile network environment. Nodes in particular areas are connected to hubs (creating stars), and the hubs are connected together along the network backbone (like a bus network). Quite often, stars are nested within stars, as seen in the example below:



A typical star bus network

- **Local Area Network (LAN)** - A LAN is a network of computers that are in the same general physical location, usually within a building or a campus. If the computers are far apart (such as across town or in different cities), then a **Wide Area Network (WAN)** is typically used.
- **Network Interface Card (NIC)** - Every computer (and most other devices) is connected to a network through an NIC. In most desktop computers, this is an [Ethernet](#) card (normally 10 or 100 Mbps) that is plugged into a slot on the computer's [motherboard](#).
- **Media Access Control (MAC) address** - This is the *physical* address of any device -- such as the NIC in a computer -- on the network. The MAC address has two parts, each 3 [bytes](#) long. The first 3 bytes identify the company that made the NIC. The second 3 bytes are the serial number of the NIC itself.
- **Unicast** - A unicast is a transmission from one node addressed specifically to another node.
- **Multicast** - In a multicast, a node sends a packet addressed to a special group address. Devices that are interested in this group register to receive packets addressed to the group. An example might be a [Cisco](#) router sending out an update to all of the other Cisco routers.
- **Broadcast** - In a broadcast, a node sends out a packet that is intended for transmission to all other nodes on the network.

## Adding Switches

In the most basic type of network found today, nodes are simply connected together using hubs. As a network grows, there are some potential problems with this configuration:

- **Scalability** - In a hub network, limited shared bandwidth makes it difficult to accommodate significant growth without sacrificing performance. Applications today need more bandwidth than ever before. Quite often, the entire network must be redesigned periodically to accommodate growth.
- **Latency** - This is the amount of time that it takes a [packet](#) to get to its destination. Since each node in a hub-based network has to wait for an opportunity to transmit in order to avoid **collisions**, the latency can increase significantly as you add more nodes. Or, if someone is transmitting a large file across the network, then all of the other nodes have to wait for an opportunity to send their own packets. You have probably seen this before at work -- you try to access a server or the Internet and suddenly everything slows down to a crawl.
- **Network failure** - In a typical network, one device on a hub can cause problems for other devices attached to the hub due to incorrect speed settings (100 Mbps on a 10-Mbps hub) or excessive broadcasts. Switches can be configured to limit broadcast levels.
- **Collisions** - Ethernet uses a process called **CSMA/CD** (Carrier Sense Multiple Access with Collision Detection) to communicate across the network. Under CSMA/CD, a node will not send out a packet unless the network is clear of traffic. If two nodes send out packets at the same time, a collision occurs and the packets are lost. Then both nodes wait a random amount of time and retransmit the packets. Any part of the network where there is a possibility that packets from two or more nodes will interfere with each other is considered to be part of the same **collision domain**. A network with a large number of nodes on the same segment will often have a lot of collisions and therefore a large collision domain.

While hubs provide an easy way to scale up and shorten the distance that the packets must travel to get from one node to another, they do not break up the actual network into discrete segments. That is where switches come in.



**Imagine that each vehicle is a packet of data waiting for an opportunity to continue on its trip.**

Think of a hub as a four-way intersection where everyone has to stop. If more than one car reaches the intersection at the same time, they have to wait for their turn to proceed. Now imagine what this would be like with a dozen or even a hundred roads intersecting at a single point. The amount of waiting and the potential for a collision increases significantly. But wouldn't it be amazing if you could take an exit ramp from any one of those roads to the road of your choosing? That is exactly what a switch does for network traffic. A switch is like a cloverleaf intersection -- each car can take an exit ramp to get to its destination without having to stop and wait for other traffic to go by.

A vital difference between a hub and a switch is that all the nodes connected to a hub share the bandwidth among themselves, while a device connected to a switch port has the **full bandwidth** all to itself. For example, if 10 nodes are communicating using a hub on a 10-Mbps network, then each node may only get a portion of the 10 Mbps if other nodes on the hub want to communicate as well. But with a switch, each node could possibly communicate at the full 10 Mbps. Think about our road analogy. If all of the traffic is coming to a common intersection, then each car has to share that intersection with every other car. But a cloverleaf allows all of the traffic to continue at full speed from one road to the next.

In a **fully switched network**, switches replace all the hubs of an Ethernet network with a dedicated segment for every node. These segments connect to a switch, which supports multiple dedicated segments (sometimes in the hundreds). Since the only devices on each segment are the switch and the node, the switch picks up every transmission before it reaches another node. The switch then forwards the frame over the appropriate segment. Since any segment contains only a single node, the frame only reaches the intended recipient. This allows many conversations to occur simultaneously on a switched network.

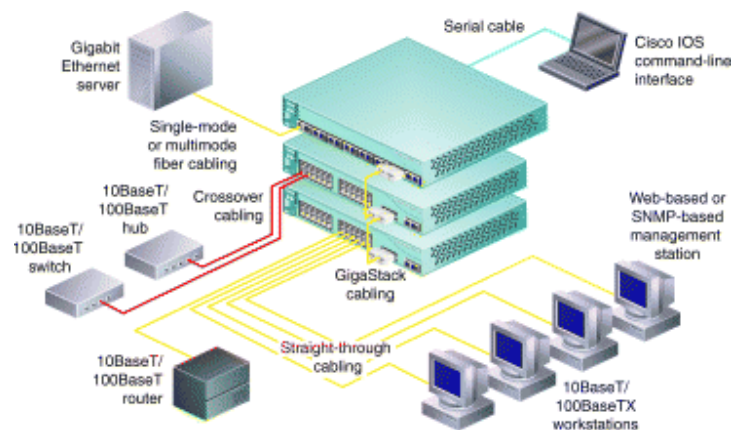
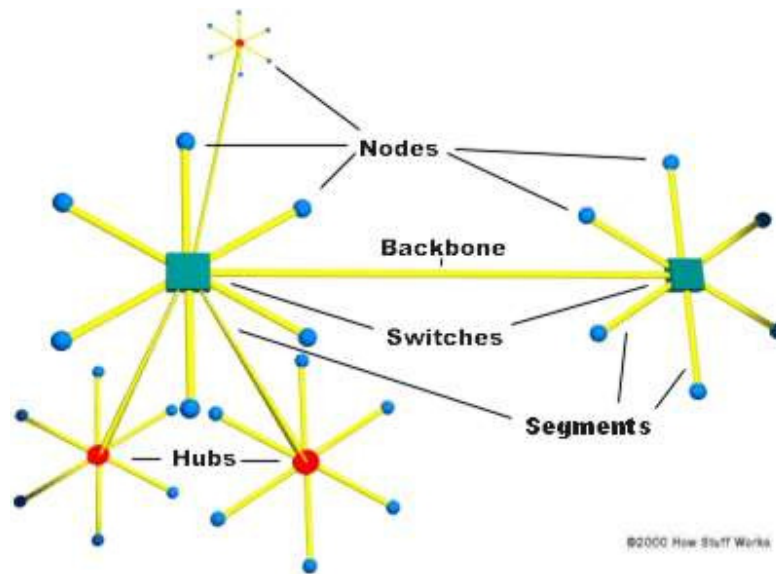


Image courtesy Cisco Networks

**An example of a network using a switch**

Switching allows a network to maintain **full-duplex** Ethernet. Before switching, Ethernet was half-duplex, which means that data could be transmitted in only one direction at a time. In a fully switched network, each node communicates only with the switch, not directly with other nodes. Information can travel from node to switch and from switch to node simultaneously.

Fully switched networks employ either twisted-pair or fiber-optic cabling, both of which use separate conductors for sending and receiving data. In this type of environment, Ethernet nodes can forgo the collision detection process and transmit at will, since they are the only potential devices that can access the medium. In other words, traffic flowing in each direction has a lane to itself. This allows nodes to transmit to the switch as the switch transmits to them -- it's a collision-free environment. Transmitting in both directions can effectively double the apparent speed of the network when two nodes are exchanging information. If the speed of the network is 10 Mbps, then each node can transmit simultaneously at 10 Mbps.



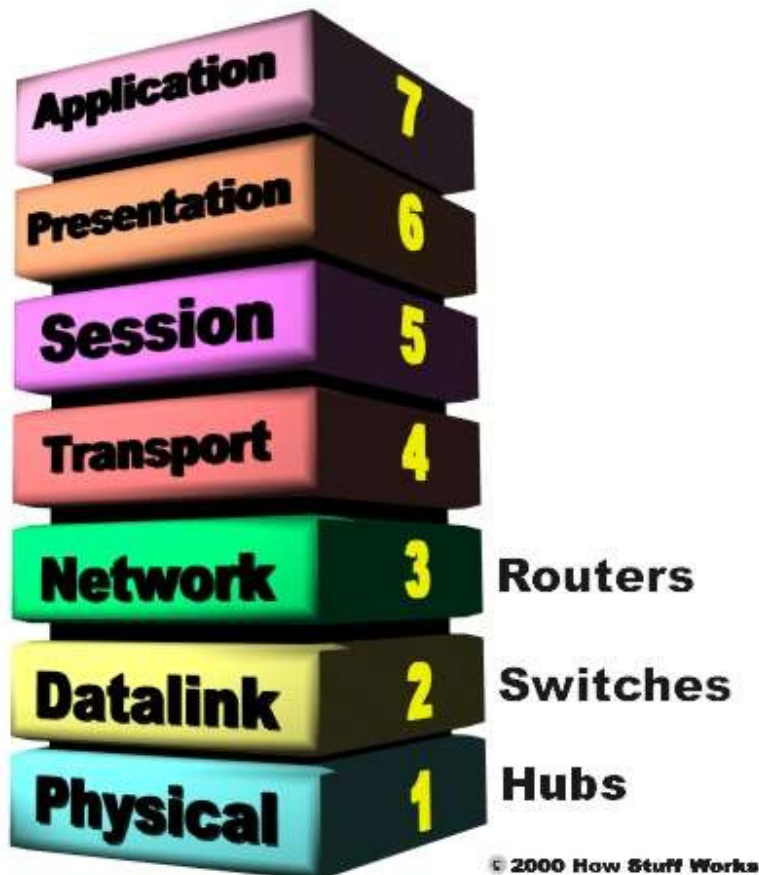
**A mixed network with two switches and three hubs**

Most networks are not fully switched because of the costs incurred in replacing all of the hubs with switches. Instead, a combination of switches and hubs are used to create an efficient yet cost-effective network. For example, a company may have hubs connecting the computers in each department and then a switch connecting all of the department-level hubs.

## Switching Technologies

You can see that a switch has the potential to radically change the way nodes communicate with each other. But you may be wondering what makes it different from a [router](#). Switches usually work at [Layer 2](#) (Data or Datalink) of the [OSI Reference Model](#), using MAC addresses, while routers work at [Layer 3](#) (Network) with Layer 3 addresses (IP, IPX or Appletalk, depending on which [Layer 3 protocols](#) are being used). The [algorithm](#) that switches use to decide how to forward packets is different from the algorithms used by routers to forward packets.

One of these differences in the algorithms between switches and routers is how **broadcasts** are handled. On any network, the concept of a broadcast packet is vital to the operability of a network. Whenever a device needs to send out information but doesn't know who it should send it to, it sends out a broadcast. For example, every time a new computer or other device comes on to the network, it sends out a broadcast packet to announce its presence. The other nodes (such as a [domain server](#)) can add the computer to their **browser list** (kind of like an address directory) and communicate directly with that computer from that point on. Broadcasts are used any time a device needs to make an announcement to the rest of the network or is unsure of who the recipient of the information should be.



The OSI Reference Model consists of seven layers that build from the wire (Physical) to the software (Application).

A hub or a switch will pass along any broadcast packets they receive to all the other segments in the broadcast domain, but a router will not. Think about our four-way intersection again: All of the traffic passed through the intersection no matter where it was going. Now imagine that this intersection is at an international border. To pass through the intersection, you must provide the border guard with the specific address that you are going to. If you don't have a specific destination, then the guard will not let you pass. A router works like this. Without the specific address of another device, it will not let the data packet through. This is a good thing for keeping networks separate from each other, but not so good when you want to talk between different parts of the same network. This is where switches come in.

LAN switches rely on **packet-switching**. The switch establishes a connection between two segments just long enough to send the current packet. Incoming packets (part of an Ethernet **frame**) are saved to a temporary memory area (**buffer**); the MAC address contained in the frame's **header** is read and then compared to a list of addresses maintained in the switch's **lookup table**. In an Ethernet-based LAN, an Ethernet frame contains a normal packet as the **payload** of the frame, with a special header that includes the MAC address information for the source and destination of the packet.

Packet-based switches use one of three methods for routing traffic:

- **Cut-through**
- **Store-and-forward**
- **Fragment-free**

**Cut-through** switches read the MAC address as soon as a packet is detected by the switch. After storing the 6 bytes that make up the address information, they immediately begin sending the packet to the destination node, even as the rest of the packet is coming into the switch.

A switch using **store-and-forward** will save the entire packet to the buffer and check it for **CRC** errors or other problems before sending. If the packet has an error, it is discarded. Otherwise, the switch looks up the MAC address and sends the packet on to the destination node. Many switches combine the two methods, using cut-through until a certain error level is reached and then changing over to store-and-forward. Very few switches are strictly cut-through, since this provides no error correction.

A less common method is **fragment-free**. It works like cut-through except that it stores the first 64 bytes of the packet before sending it on. The reason for this is that most errors, and all collisions, occur during the initial 64 bytes of a packet.

LAN switches vary in their physical design. Currently, there are three popular configurations in use:

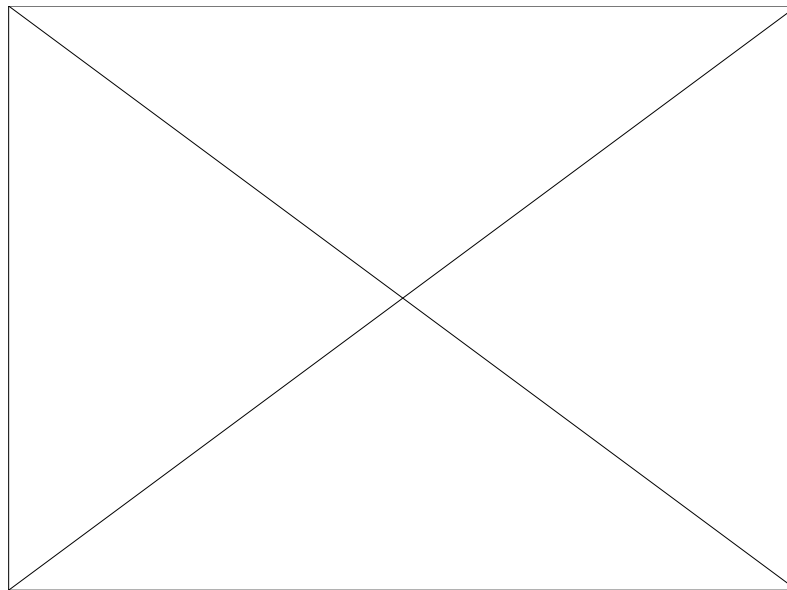
- **Shared memory** - This type of switch stores all incoming packets in a common memory buffer shared by all the switch **ports** (input/output connections), then sends them out via the correct port for the destination node.
- **Matrix** - This type of switch has an internal grid with the input ports and the output ports crossing each other. When a packet is detected on an input port, the MAC address is compared to the lookup table to find the appropriate output port. The switch then makes a connection on the grid where these two ports intersect.
- **Bus architecture** - Instead of a grid, an internal transmission path (**common bus**) is shared by all of the ports using [TDMA](#). A switch based on this configuration has a dedicated memory buffer for each port, as well as an [ASIC](#) to control the internal bus access.

## Transparent Bridging

Most Ethernet LAN switches use a very cool system called **transparent bridging** to create their address lookup tables. Transparent bridging is a technology that allows a switch to learn everything it needs to know about the location of nodes on the network without the network administrator having to do anything. Transparent bridging has five parts:

- **Learning**
- **Flooding**
- **Filtering**
- **Forwarding**
- **Aging**

Here's how it works:



**Click on the menu terms to learn more about how transparent bridging works.**

- The switch is added to the network, and the various segments are plugged into the switch's ports.
- A computer (Node A) on the first segment (Segment A) sends data to a computer (Node B) on another segment (Segment C).
- The switch gets the first packet of data from Node A. It reads the MAC address and saves it to the lookup table for Segment A. The switch now knows where to find Node A anytime a packet is addressed to it. This process is called **learning**.
- Since the switch does not know where Node B is, it sends the packet to all of the segments except the one that it arrived on (Segment A). When a switch sends a packet out to all segments to find a specific node, it is called **flooding**.
- Node B gets the packet and sends a packet back to Node A in acknowledgement.
- The packet from Node B arrives at the switch. Now the switch can add the MAC address of Node B to the lookup table for Segment C. Since the switch already knows the address of Node A, it sends the packet directly to it. Because Node A is on a different segment than Node B, the

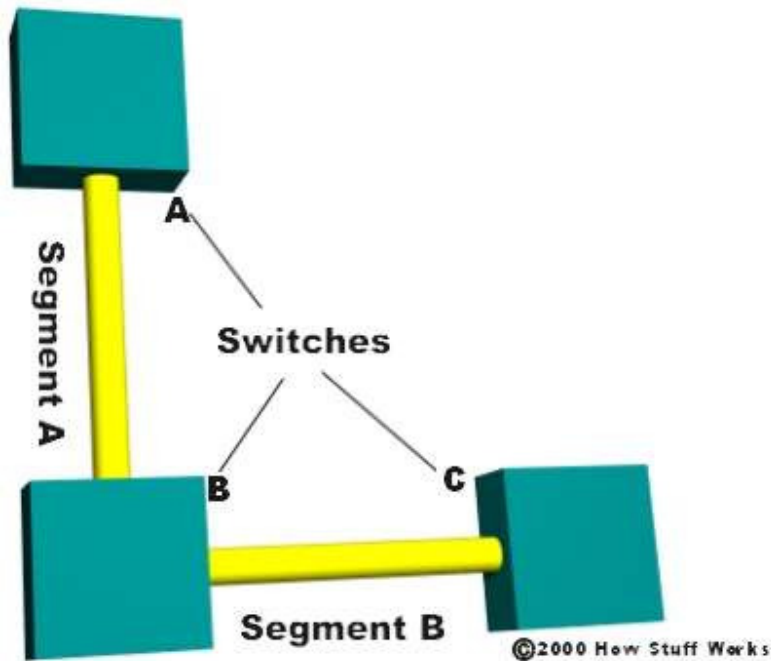
switch must connect the two segments to send the packet. This is known as **forwarding**.

- The next packet from Node A to Node B arrives at the switch. The switch now has the address of Node B, too, so it forwards the packet directly to Node B.
- Node C sends information to the switch for Node A. The switch looks at the MAC address for Node C and adds it to the lookup table for Segment A. The switch already has the address for Node A and determines that both nodes are on the same segment, so it does not need to connect Segment A to another segment for the data to travel from Node C to Node A. Therefore, the switch will ignore packets traveling between nodes on the same segment. This is **filtering**.
- Learning and flooding continue as the switch adds nodes to the lookup tables. Most switches have plenty of **memory** in a switch for maintaining the lookup tables; but to optimize the use of this memory, they still remove older information so that the switch doesn't waste time searching through stale addresses. To do this, switches use a technique called **aging**. Basically, when an entry is added to the lookup table for a node, it is given a timestamp. Each time a packet is received from a node, the timestamp is updated. The switch has a user-configurable timer that erases the entry after a certain amount of time with no activity from that node. This frees up valuable memory resources for other entries. As you can see, transparent bridging is a great and essentially maintenance-free way to add and manage all the information a switch needs to do its job!

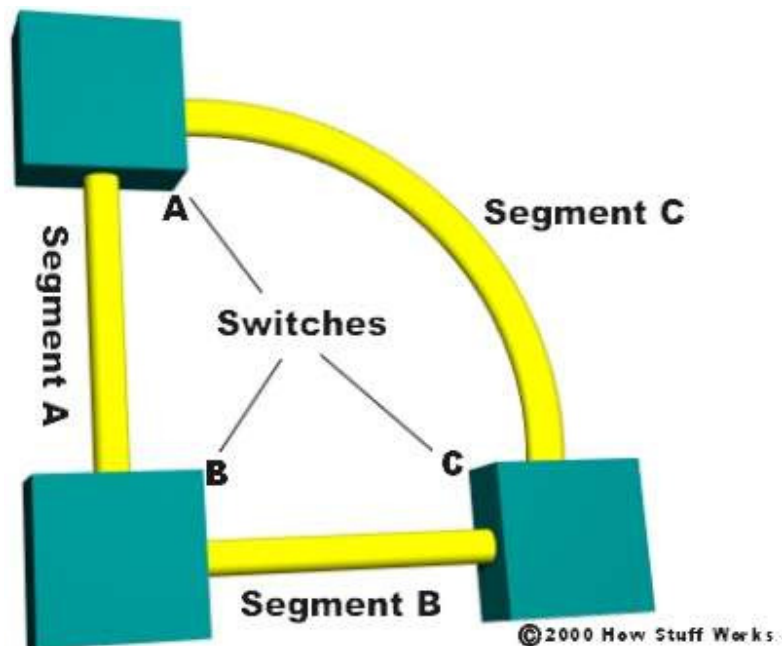
In our example, two nodes share segment A, while the switch creates independent segments for Node B and Node D. In an ideal LAN-switched network, every node would have its own segment. This would eliminate the possibility of collisions and also the need for filtering.

## Redundancy and Broadcast Storms

When we talked about bus and ring networks earlier, one issue was the possibility of a single point of failure. In a star or star-bus network, the point with the most potential for bringing all or part of the network down is the switch or hub. Look at the example below:

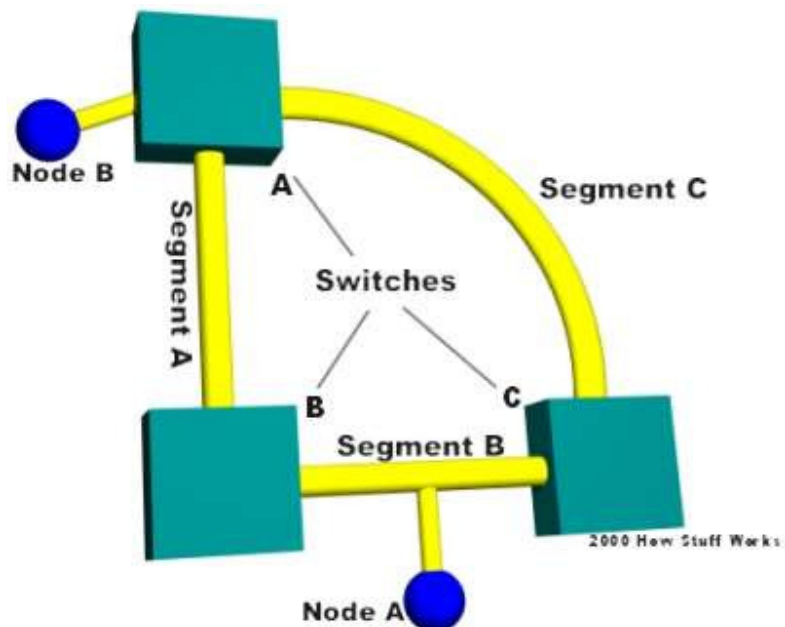


In this example, if either switch A or C fails, then the nodes connected to that particular switch are affected, but nodes at the other two switches can still communicate. However, if switch B fails, then the entire network is brought down. What if we add another segment to our network connecting switches A and C?



In this case, even if one of the switches fails, the network will continue. This provides **redundancy**, effectively eliminating the single point of failure.

But now we have a new problem. In the last section, you discovered how switches learn where the nodes are located. With all of the switches now connected in a loop, a packet from a node could quite possibly come to a switch from two different segments. For example, imagine that Node B is connected to Switch A, and needs to communicate with Node A on Segment B. Switch A does not know who Node A is, so it floods the packet.



The packet travels via Segment A or Segment C to the other two switches (B and C). Switch B will add Node B to the lookup table it maintains for Segment A, while Switch C will add it to the lookup table for Segment C. If neither switch has learned the address for Node A yet, they will flood Segment B looking for Node A. Each switch will take the packet sent by the other switch and flood it back out again immediately, since they still don't know who Node A is. Switch A will receive the packet from each segment and flood it back out on the other segment. This causes a **broadcast storm** as the packets are broadcast, received and rebroadcast by each switch, resulting in potentially severe network congestion.

Which brings us to **spanning trees**...

## Spanning Trees

To prevent broadcast storms and other unwanted side effects of looping, [Digital Equipment Corporation](#) created the **spanning-tree protocol** (STP), which has been standardized as the **802.1d** specification by the [Institute of Electrical and Electronic Engineers](#) (IEEE). Essentially, a spanning tree uses the **spanning-tree algorithm** (STA), which senses that the switch has more than one way to communicate with a node, determines which way is best and blocks out the other path(s). The cool thing is that it keeps track of the other path(s), just in case the primary path is unavailable.

Here's how STP works:

- Each switch is assigned a group of IDs, one for the switch itself and one for each port on the switch. The switch's identifier, called the **bridge ID** (BID), is 8 bytes long and contains a bridge priority (2 bytes) along with one of the switch's MAC addresses (6 bytes). Each **port ID** is 16 bits long with two parts: a 6-bit priority setting and a 10-bit port number.
- A **path cost** value is given to each port. The cost is typically based on a guideline established as part of 802.1d. According to the original specification, cost is 1,000 Mbps (1 gigabit per second) divided by the bandwidth of the segment connected to the port. Therefore, a 10 Mbps connection would have a cost of  $(1,000/10)$  100.

To compensate for the speed of networks increasing beyond the gigabit range, the standard cost has been slightly modified. The new cost values are:

Bandwidth	STP Cost Value
4 Mbps	250
10 Mbps	100
16 Mbps	62
45 Mbps	39
100 Mbps	19
155 Mbps	14
622 Mbps	6
1 Gbps	4
10 Gbps	2

You should also note that the path cost can be an arbitrary value assigned by the network administrator, instead of one of the standard cost values.

- Each switch begins a discovery process to choose which network paths it should use for each segment. This information is shared between all the switches by way of special network frames called **bridge protocol data units** (BPDU). The parts of a BPDU are:
  - **Root BID** - This is the BID of the current **root bridge**.
  - **Path cost to root bridge** - This determines how far away the root bridge is. For example, if the data has to travel over three 100-Mbps segments to reach the root bridge, then the cost is  $(19 + 19 + 0)$  38. The segment attached to the root bridge will normally have a path cost of zero.
  - **Sender BID** - This is the BID of the switch that sends the BPDU.
  - **Port ID** - This is the actual port on the switch that the BPDU was sent from.

All of the switches are constantly sending BPDUs to each other, trying to determine the best path between various segments. When a switch receives a BPDU (from another switch) that is better than the one it is broadcasting for the same segment, it will stop broadcasting its BPDU out that segment. Instead, it will store the other switch's BPDU for reference and for broadcasting out to **inferior segments**, such as those that are farther away from the root bridge.

- A **root bridge** is chosen based on the results of the BPDU process between the switches. Initially, every switch considers itself the root bridge. When a switch first powers up on the network, it sends out a BPDU with its own BID as the root BID. When the other switches receive the BPDU, they compare the BID to the one they already have stored as the root BID. If the new root BID has a lower value, they replace the saved one. But if the saved root BID is lower, a BPDU is sent to the new switch with this BID as the root BID. When the new switch receives the BPDU, it realizes that it is not the root bridge and replaces the root BID in its table with the one it just received. The result is that the switch that has the lowest BID is elected by the other switches as the root bridge.
- Based on the location of the root bridge, the other switches determine which of their ports has the lowest path cost to the root bridge. These ports are called **root ports**, and each switch (other than the current root bridge) must have one.

- The switches determine who will have **designated ports**. A designated port is the connection used to send and receive packets on a specific segment. By having only one designated port per segment, all looping issues are resolved!

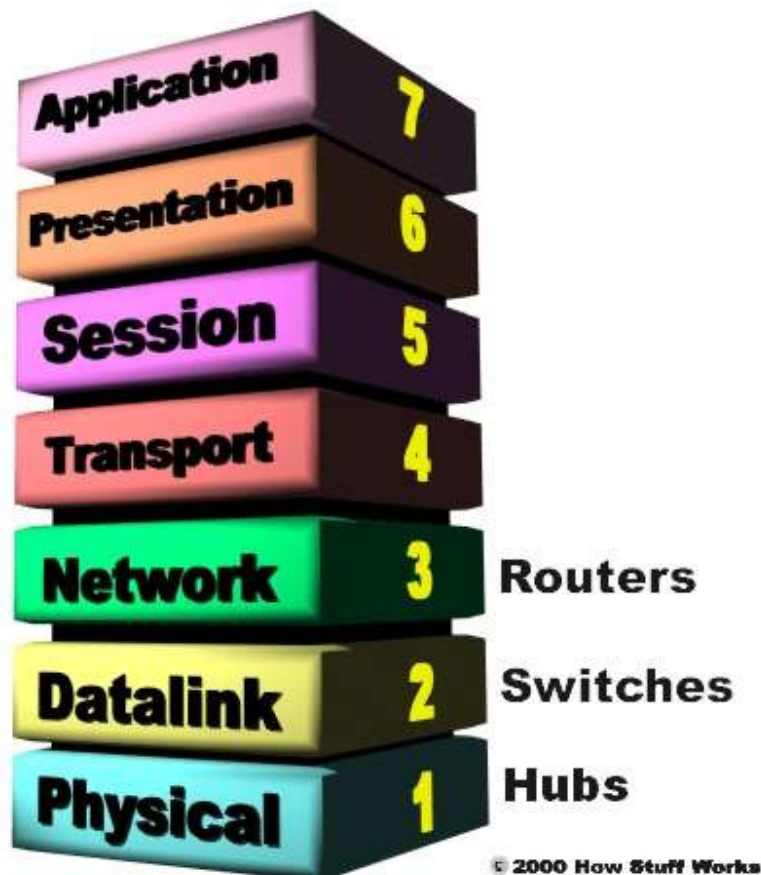
Designated ports are selected based on the lowest path cost to the root bridge for a segment. Since the root bridge will have a path cost of "0," any ports on it that are connected to segments will become designated ports. For the other switches, the path cost is compared for a given segment. If one port is determined to have a lower path cost, it becomes the designated port for that segment. If two or more ports have the same path cost, then the switch with the lowest BID is chosen.

- Once the designated port for a network segment has been chosen, any other ports that connect to that segment become **non-designated ports**. They block network traffic from taking that path so it can only access that segment through the designated port.

Each switch has a table of BPDUs that it continually updates. The network is now configured as a single spanning tree, with the root bridge as the trunk and all the other switches as branches. Each switch communicates with the root bridge through the root ports, and with each segment through the designated ports, thereby maintaining a loop-free network. In the event that the root bridge begins to fail or have network problems, STP allows the other switches to immediately reconfigure the network with another switch acting as root bridge. This amazing process gives a company the ability to have a complex network that is fault-tolerant and yet fairly easy to maintain.

## Routers and Layer 3 Switching

While most switches operate at the **Data layer** (Layer 2) of the [OSI Reference Model](#), some incorporate features of a [router](#) and operate at the **Network layer** (Layer 3) as well. In fact, a Layer 3 switch is incredibly similar to a router.



**Layer 3 switches actually work at the Network layer.**

When a router receives a packet, it looks at the Layer 3 source and destination addresses to determine the path the packet should take. A standard switch relies on the MAC addresses to determine the source and destination of a packet, which is Layer 2 (Data) networking.

The fundamental difference between a router and a Layer 3 switch is that Layer 3 switches have optimized hardware to pass data as fast as Layer 2 switches, yet they make decisions on how to transmit traffic at Layer 3, just like a router. Within the LAN environment, a Layer 3 switch is usually faster than



The VLANs can communicate with each other via the trunking connection between the two switches using the router. For example, data from a computer on VLAN A that needs to get to a computer on VLAN B (or VLAN C or VLAN D) must travel from the switch to the router and back again to the switch. Because of the transparent bridging algorithm and trunking, both PCs and the router think that they are on the same physical segment!

As you can see, LAN switches are an amazing technology that can really make a difference in the speed and quality of a network.

For more information, check out the links on the next page.

## Lots More Information!

### Related HowStuffWorks Articles

- [How Home Networking Works](#)
- [How Routers Work](#)
- [How Ethernet Works](#)
- [How OSI Works](#)
- [How Firewalls Work](#)
- [How Network Address Translation Works](#)
- [How Web Servers Work](#)
- [How Virtual Private Networks Work](#)
- [What is a packet?](#)
- [What is an IP address?](#)

### More Great Links

- [Webopedia: switch](#)
- [Cisco: Internetworking Technology Overview](#)
- [Cirrus: LAN Switch](#)
- [University of New Hampshire InterOperability Lab: Ethernet Tutorials and Resources](#)
- [Cisco: Understanding Spanning-Tree Protocol](#)
- [Cisco VLAN Roadmap](#)
- [VLAN tagging for linux](#)
- [Are there Vulnerabilities in VLAN Implementations?](#)
- [Ethernet Media Access Control](#)
- [Full Duplex Ethernet & Fiber Optic Cabling](#)
- [Layer 3 Switching Demystified](#)

## How does a T1 line work?

Most of us are familiar with a normal business or residential line from the phone company. A normal phone line like this is delivered on a pair of copper wires (see [How Telephones Work](#)) that transmit your voice as an analog signal (see [How Analog and Digital Recording Works](#) for details). When you use a normal modem on a line like this, it can transmit data at perhaps 30 kilobits per second (30,000 bits per second).

The phone company moves nearly all voice traffic as digital rather than analog signals. Your analog line gets converted to a digital signal by sampling it 8,000 times per second at 8-bit resolution (64,000 bits per second). Nearly all digital data now flows over fiber optic lines, and the phone company uses different designations to talk about the capacity of a fiber optic line.

If your office has a T1 line, it means that the phone company has brought a fiber optic line into your office (a T1 line might also come in on copper). A T1 line can carry 24 digitized voice channels, or it can carry data at a rate of 1.544 megabits per second. If the T1 line is being used for telephone conversations, it plugs into the office's phone system. If it is carrying data it plugs into the network's router.

A T1 line can carry about 192,000 bytes per second -- roughly 60 times more data than a normal residential modem. It is also extremely reliable -- much more reliable than an analog modem. Depending on what they are doing, a T1 line can generally handle quite a few people. For general browsing, hundreds of users are easily able to share a T1 line comfortably. If they are all downloading MP3 files or video files simultaneously it would be a problem, but that still isn't extremely common.

A T1 line might cost between \$1,000 and \$1,500 per month depending on who provides it and where it goes. The other end of the T1 line needs to be connected to an ISP (see [How the Internet Works](#)), and the total cost is a combination of the fee the phone company charges and the fee the ISP charges.

A large company needs something more than a T1 line. The following table shows some of the common line designations:

- DS0 - 64 kilobits per second
- ISDN - Two DS0 lines plus signaling (16 kilobits per second), or 128 kilobits per second
- T1 - 1.544 megabits per second (24 DS0 lines)
- T3 - 43.232 megabits per second (28 T1s)
- OC3 - 155 megabits per second (84 T1s)
- OC12 - 622 megabits per second (4 OC3s)
- OC48 - 2.5 gigabits per seconds (4 OC12s)
- OC192 - 9.6 gigabits per second (4 OC48s)

Here are three interesting links:

- [ISDN Intro](#)
- [WAN technologies](#)
- [Net Access Leased-line FAQ](#)
- [How bits and bytes work](#) -- explains kilobits, megabits, gigabits, etc.

## How Fiber Optics Work

by [Craig C. Freudenrich, Ph.D.](#)

You hear about fiber-optic cables whenever people talk about the [telephone system](#), the [cable TV system](#) or the Internet. Fiber-optic lines are strands of optically pure **glass** as thin as a human hair that carry digital information over long distances. They are also used in medical imaging and mechanical engineering inspection.

In this article, we will show you how these tiny strands of glass transmit light and the fascinating way that these strands are made.

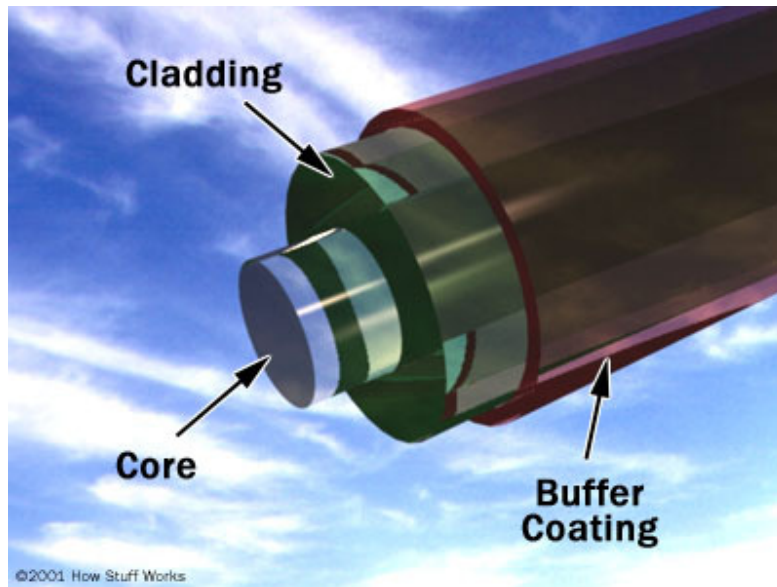
## What are Fiber Optics?

**Fiber optics** (optical fibers) are long, thin strands of very pure glass about the diameter of a human hair. They are arranged in bundles called **optical cables** and used to transmit [light](#) signals over long distances.



Photo courtesy [Corning](#)

**A fiber-optic wire**



**Parts of a single optical fiber**

If you look closely at a single optical fiber, you will see that it has the following parts:

- **Core** - Thin glass center of the fiber where the light travels
- **Cladding** - Outer optical material surrounding the core that reflects the light back into the core
- **Buffer coating** - Plastic coating that protects the fiber from damage and moisture

Hundreds or thousands of these optical fibers are arranged in bundles in optical cables. The bundles are protected by the cable's outer covering, called a **jacket**.

Optical fibers come in two types:

- **Single-mode fibers**
- **Multi-mode fibers**

See [Tpub.com: Mode Theory](http://Tpub.com: Mode Theory) for a good explanation.

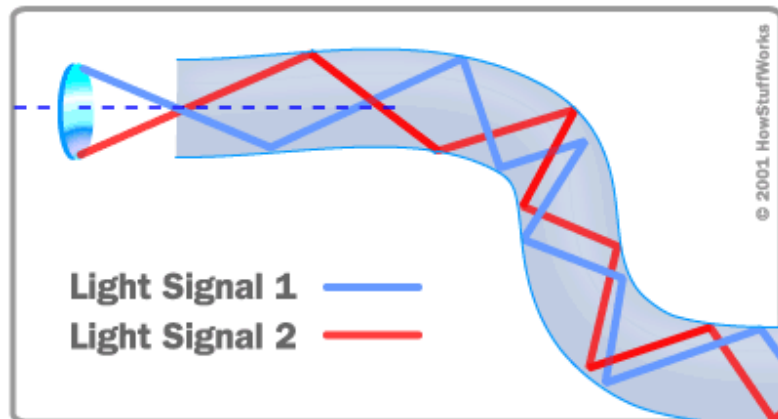
**Single-mode fibers** have small cores (about  $3.5 \times 10^{-4}$  inches or 9 microns in diameter) and transmit infrared [laser](#) light (wavelength = 1,300 to 1,550 nanometers). **Multi-mode fibers** have larger cores (about  $2.5 \times 10^{-3}$  inches or 62.5 microns in diameter) and transmit infrared light (wavelength = 850 to 1,300 nm) from [light-emitting diodes](#) (LEDs).

Some optical fibers can be made from **plastic**. These fibers have a large core (0.04 inches or 1 mm diameter) and transmit visible red light (wavelength = 650 nm) from LEDs.

Let's look at how an optical fiber works.

## How Does an Optical Fiber Transmit Light?

Suppose you want to shine a flashlight beam down a long, straight hallway. Just point the beam straight down the hallway -- light travels in straight lines, so it is no problem. What if the hallway has a bend in it? You could place a mirror at the bend to reflect the light beam around the corner. What if the hallway is very winding with multiple bends? You might line the walls with mirrors and angle the beam so that it bounces from side-to-side all along the hallway. This is exactly what happens in an optical fiber.



**Diagram of total internal reflection in an optical fiber**

The light in a fiber-optic cable travels through the core (hallway) by constantly bouncing from the cladding (mirror-lined walls), a principle called **total internal reflection**. Because the cladding does not absorb any light from the core, the light wave can travel great distances. However, some of the light signal **degrades** within the fiber, mostly due to impurities in the glass. The extent that the signal degrades depends on the purity of the glass and the wavelength of the transmitted light (for example, 850 nm = 60 to 75 percent/km; 1,300 nm = 50 to 60 percent/km; 1,550 nm is greater than 50 percent/km). Some premium optical fibers show much less signal degradation -- less than 10 percent/km at 1,550 nm.

**Need to Know More?**  
Check out a [detailed description](#) of the physics of total internal reflection.

## A Fiber-Optic Relay System

To understand how optical fibers are used in communications systems, let's look at an example from a World War II movie or documentary where two naval ships in a fleet need to communicate with each other while maintaining [radio](#) silence or on stormy seas. One ship pulls up alongside the other. The captain of one ship sends a message to a sailor on deck. The sailor translates the message into Morse code (dots and dashes) and uses a signal light (floodlight with a venetian blind type shutter on it) to send the message to the other ship. A sailor on the deck of the other ship sees the Morse code message, decodes it into English and sends the message up to the captain.

Now, imagine doing this when the ships are on either side of the ocean separated by thousands of miles and you have a fiber-optic communication system in place between the two ships. Fiber-optic relay systems consist of the following:

- **Transmitter** - Produces and encodes the light signals
- **Optical fiber** - Conducts the light signals over a distance
- **Optical regenerator** - May be necessary to boost the light signal (for long distances)
- **Optical receiver** - Receives and decodes the light signals

### Transmitter

The **transmitter** is like the sailor on the deck of the sending ship. It receives and directs the optical device to turn the light "on" and "off" in the correct sequence, thereby generating a light signal.

The transmitter is physically close to the optical fiber and may even have a lens to focus the light into the fiber. Lasers have more power than LEDs, but vary more with changes in temperature and are more expensive. The most common wavelengths of light signals are 850 nm, 1,300 nm, and 1,550 nm (infrared, non-visible portions of the [spectrum](#)).

### Optical Regenerator

As mentioned above, some **signal loss** occurs when the light is transmitted through the fiber, especially over long distances (more than a half mile, or about 1 km) such as with undersea cables. Therefore, one or more **optical regenerators** is spliced along the cable to boost the degraded light signals.

An optical regenerator consists of optical fibers with a special coating (**doping**). The doped portion is "pumped" with a [laser](#). When the degraded signal comes into the doped coating, the energy from the laser allows the doped molecules to become lasers themselves. The doped molecules then emit a new, stronger light signal with the same characteristics as the incoming weak light signal. Basically, the regenerator is a laser amplifier for the incoming signal. See [Photonics.com: Fiber Amplifiers](#) for more details.

### Optical Receiver

The **optical receiver** is like the sailor on the deck of the receiving ship. It takes the incoming digital light signals, decodes them and sends the electrical signal to the other user's [computer](#), [TV](#) or [telephone](#) (receiving ship's captain). The receiver uses a **photocell** or **photodiode** to detect the light.

# Advantages of Fiber Optics

Why are fiber-optic systems revolutionizing telecommunications? Compared to conventional metal wire (copper wire), optical fibers are:

- **Less expensive** - Several miles of optical cable can be made cheaper than equivalent lengths of copper wire. This saves your provider (cable TV, Internet) and you money.
- **Thinner** - Optical fibers can be drawn to smaller diameters than copper wire.
- **Higher carrying capacity** - Because optical fibers are thinner than copper wires, more fibers can be bundled into a given-diameter cable than copper wires. This allows more phone lines to go over the same cable or more channels to come through the cable into your cable TV box.
- **Less signal degradation** - The loss of signal in optical fiber is less than in copper wire.
- **Light signals** - Unlike electrical signals in copper wires, light signals from one fiber do not interfere with those of other fibers in the same cable. This means clearer phone conversations or TV reception.
- **Low power** - Because signals in optical fibers degrade less, lower-power transmitters can be used instead of the high-voltage electrical transmitters needed for copper wires. Again, this saves your provider and you money.
- **Digital signals** - Optical fibers are ideally suited for carrying digital information, which is especially useful in computer networks.
- **Non-flammable** - Because no electricity is passed through optical fibers, there is no fire hazard.
- **Lightweight** - An optical cable weighs less than a comparable copper wire cable. Fiber-optic cables take up less space in the ground.
- **Flexible** - Because fiber optics are so flexible and can transmit and receive light, they are used in many flexible [digital cameras](#) for the following purposes:
  - **Medical imaging** - in bronchoscopes, endoscopes, laparoscopes
  - **Mechanical imaging** - inspecting mechanical welds in pipes and engines (in [airplanes](#), [rockets](#), [space shuttles](#), [cars](#))
  - **Plumbing** - to inspect [sewer lines](#)

Because of these advantages, you see fiber optics in many industries, most notably telecommunications and computer networks. For example, if you telephone Europe from the United States (or vice versa) and the signal is bounced off a communications [satellite](#), you often hear an echo on the line. But with transatlantic fiber-optic cables, you have a direct connection with no echoes.

## How Are Optical Fibers Made?

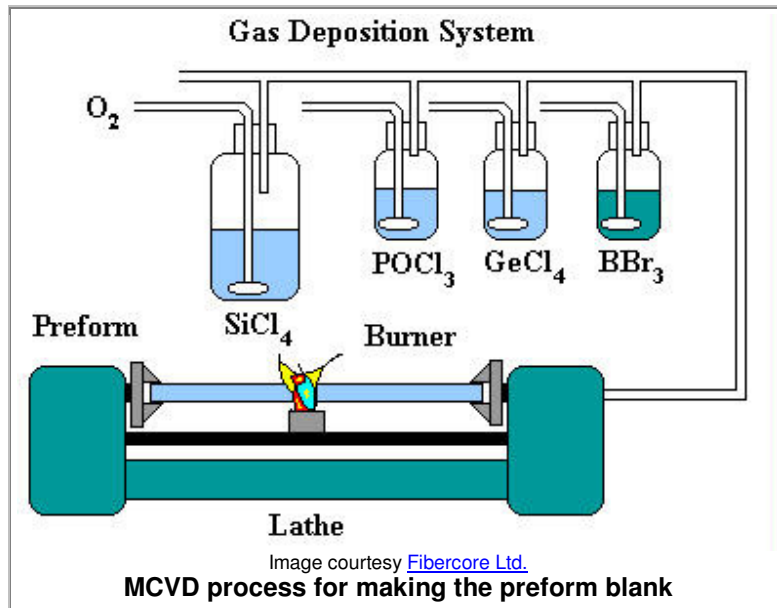
Now that we know how fiber-optic systems work and why they are useful -- how do they make them? Optical fibers are made of extremely pure **optical glass**. We think of a glass window as transparent, but the thicker the glass gets, the less transparent it becomes due to impurities in the glass. However, the glass in an optical fiber has far fewer impurities than window-pane glass. One company's description of the quality of glass is as follows: If you were on top of an ocean that is miles of solid core optical fiber glass, you could see the bottom clearly.

Making optical fibers requires the following steps:

1. **Making a preform glass cylinder**
2. **Drawing the fibers from the preform**
3. **Testing the fibers**

## Making the Preform Blank

The glass for the preform is made by a process called **modified chemical vapor deposition** (MCVD).



In MCVD, oxygen is bubbled through solutions of silicon chloride ( $\text{SiCl}_4$ ), germanium chloride ( $\text{GeCl}_4$ ) and/or other chemicals. The precise mixture governs the various physical and optical properties (index of refraction, coefficient of expansion, melting point, etc.). The gas vapors are then conducted to the inside of a **synthetic silica or quartz tube** (cladding) in a special **lathe**. As the lathe turns, a torch is moved up and down the outside of the tube. The extreme heat from the torch causes two things to happen:

- The silicon and germanium react with oxygen, forming silicon dioxide ( $\text{SiO}_2$ ) and germanium dioxide ( $\text{GeO}_2$ ).
- The silicon dioxide and germanium dioxide deposit on the inside of the tube and fuse together to form glass.

The lathe turns continuously to make an even coating and consistent blank. The purity of the glass is maintained by using corrosion-resistant plastic in the gas delivery system (valve blocks, pipes, seals) and by precisely controlling the flow and composition of the mixture. The process of making the preform blank is highly automated and takes several hours. After the preform blank cools, it is tested for quality control ([index of refraction](#)).

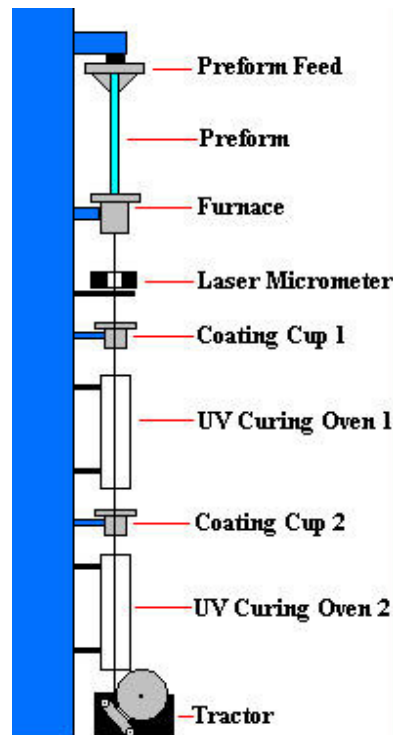


Photo courtesy Fibercore Ltd.

### Lathe used in preparing the preform blank

## Drawing Fibers from the Preform Blank

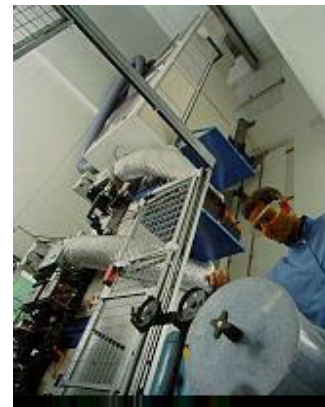
Once the preform blank has been tested, it gets loaded into a **fiber drawing tower**.



**Diagram of a fiber drawing tower used to draw optical glass fibers from a preform blank**

The blank gets lowered into a graphite furnace (3,452 to 3,992 degrees Fahrenheit or 1,900 to 2,200 degrees Celsius) and the tip gets melted until a molten glob falls down by [gravity](#). As it drops, it cools and forms a thread.

The operator threads the strand through a series of coating cups (buffer coatings) and ultraviolet light curing ovens onto a tractor-controlled spool. The tractor mechanism slowly pulls the fiber from the heated preform blank and is precisely controlled by using a **laser micrometer** to measure the diameter of the fiber and feed the information back to the tractor mechanism. Fibers are pulled from the blank at a rate of 33 to 66 ft/s (10 to 20 m/s) and the finished product is wound onto the spool. It is not uncommon for spools to contain more than 1.4 miles (2.2 km) of optical fiber.



## Testing the Finished Optical Fiber

The finished optical fiber is tested for the following:

- **Tensile strength** - Must withstand 100,000 lb/in<sup>2</sup> or more
- **Refractive index profile** - as screen for optical defects
- **Fiber geometry** - Core coating diameter are
- **Attenuation** - Determine wavelengths degrade over
- **Information carrying** signals that can be carried
- **Chromatic dispersion** - light through the core (important for bandwidth)
- **Operating temperature/humidity range**
- **Temperature dependence of attenuation**
- **Ability to conduct light underwater** - Important for undersea cables



Photo courtesy Corning  
**Finished spool of optical fiber**

Determine numerical aperture as well

diameter, cladding dimensions and uniform

the extent that light signals of various distance

**capacity** (bandwidth) - Number of at one time (multi-mode fibers)

Spread of various wavelengths of

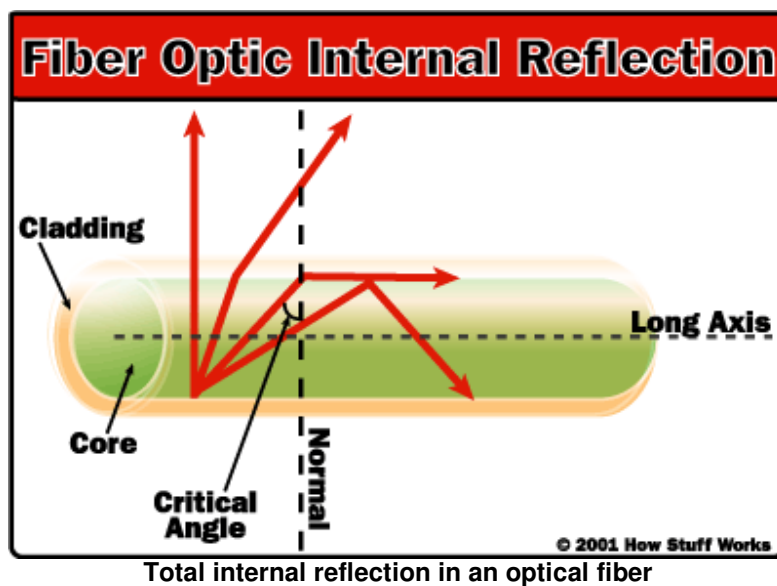
Once the fibers have passed the quality control, they are sold to telephone companies, cable companies and network providers. Many companies are currently replacing their old copper-wire-based systems with new fiber-optic-based systems to improve speed, capacity and clarity.

## Physics of Total Internal Reflection

When light passes from a medium with one [index of refraction](#) ( $m_1$ ) to another medium with a lower index of refraction ( $m_2$ ), it bends or [refracts](#) away from an imaginary line perpendicular to the surface (**normal line**). As the angle of the beam through  $m_1$  becomes greater with respect to the normal line, the refracted light through  $m_2$  bends further away from the line.

At one particular angle (**critical angle**), the refracted light will not go into  $m_2$ , but instead will travel along the surface between the two media ( **$\sin$  [critical angle] =  $n_2/n_1$**  where  $n_1$  and  $n_2$  are the indices of refraction [ $n_1$  is less than  $n_2$ ]). If the beam through  $m_1$  is greater than the critical angle, then the refracted beam will be reflected entirely back into  $m_1$  (total internal reflection), even though  $m_2$  may be transparent!

In physics, the critical angle is described with respect to the normal line. In fiber optics, the critical angle is described with respect to the parallel axis running down the middle of the fiber. Therefore, the fiber-optic critical angle = (90 degrees - physics critical angle).



In an optical fiber, the light travels through the core ( $m_1$ , high index of refraction) by constantly reflecting from the cladding ( $m_2$ , lower index of refraction) because the angle of the light is always greater than the critical angle. Light reflects from the cladding no matter what angle the fiber itself gets bent at, even if it's a full circle!

Because the cladding does not absorb any light from the core, the light wave can travel great distances. However, some of the light signal degrades within the fiber, mostly due to impurities in the glass. The extent that the signal degrades depends upon the purity of the glass and the wavelength of the transmitted light (for example, 850 nm = 60 to 75 percent/km; 1,300 nm = 50 to 60 percent/km; 1,550 nm is greater than 50 percent/km). Some premium optical fibers show much less signal degradation -- less than 10 percent/km at 1,550 nm.

For more information on fiber optics and related topics, check out the links on the next page.

## Lots More Information

### Related HowStuffWorks Articles

- [How Light Works](#)
- [How Lasers Work](#)
- [How Cable Television Works](#)
- [How DSL Works](#)

- [How LAN Switches Work](#)
- [How FireWire Works](#)
- [How Ethernet Works](#)
- [How Routers Work](#)
- [How Telephones Work](#)
- [How Web Servers Work](#)
- [How Cable Modems Work](#)
- [How Digital Clocks Work](#)
- [How does a long distance call work?](#)
- [How does a T1 line work?](#)
- [What do the little boxes that the phone company has around our neighborhood do?](#)

## More Great Links

- [Corning Optical Fiber](#)
- [Communications Specialties: Introduction to Fiber Optics](#)
- [StarTech.com: What are Fiber Optics?](#)
- [Fiberoptics Online](#)
- [Fiberoptic Product News Online](#)
- [Schott Fiber Optics: Introduction to fiber optic imaging](#)
- [Fibercore: Virtual Facilities Tour](#)

### **Total Internal Reflection**

- [What is Total Internal Reflection \(TIR\)](#)
- [The Physics Classroom: Total Internal Reflection](#)
- [Refraction: Total internal reflection](#)