

Troubleshoot LAN Switching Environments

Introduction

This document describes common LAN switch features and how to troubleshoot any LAN switching problems.

Prerequisites

Requirements

There are no specific requirements for this document.

Conventions

Refer to [Cisco Technical Tips Conventions](#) for more information on document conventions.

Background Information

The sections in this chapter describe common LAN switch features and solutions to some of the most common LAN switching problems. These topics are covered:

LAN Switching Introduction

Suggestions to troubleshoot general switch

Troubleshoot port connectivity problems

Troubleshoot Ethernet 10/100Mb half/full duplex auto-negotiation

ISL trunking on Catalyst 5000 and 6000 family switches

Configure and troubleshoot EtherChannel switch to switch

Use Portfast and other commands to fix end-station startup connectivity problems

Configure and troubleshoot multilayer switching

LAN Switching Introduction

If you are new to LAN switching, these sections take you through some of the main concepts related to switches. One of the prerequisites to troubleshoot any device is to know the rules under which it operates. Switches have become much more complex over the last few years because

they have gained in popularity and sophistication. These paragraphs describe some of the key concepts to know about switches.

Hubs and Switches

Because of the great demand placed on local area networks, there has been a shift from a shared bandwidth network, with hubs and coaxial cable, to a dedicated bandwidth network, with switches. A hub allows multiple devices to be connected to the same network segment. The devices on that segment share the bandwidth with each other. If it is a 10Mb hub, and there are 6 devices connected to 6 different ports on the hub, all six devices share the 10Mb of bandwidth with each other. A 100Mb hub shares 100Mb of bandwidth among the connected devices. In terms of the OSI model, a hub is considered a layer-one (physical layer) device. It hears an electrical signal on the wire and passes it along to the other ports.

A switch can physically replace a hub in your network. A switch allows multiple devices to be connected to the same network, just like a hub does, but this is where the similarity ends. A switch allows each connected device to have dedicated bandwidth instead of shared bandwidth. The bandwidth between the switch and the device is reserved for communication to and from that device alone. Six devices connected to six different ports on a 10Mb switch each have 10Mb of bandwidth to work with, instead of shared bandwidth with the other devices. A switch can greatly increase the available bandwidth in your network, which can lead to improved network performance.

Bridges and Switches

A basic switch is considered a layer-two device. When you use the word layer, you refer to the 7-layer OSI model. A switch does not just pass electrical signals along, like a hub does; instead, it assembles the signals into a frame (layer two), and then decides what to do with the frame. A switch determines what to do with a frame when it borrows an algorithm from another common networking device: a transparent bridge. Logically, a switch acts just like a transparent bridge would, but it can handle frames much faster than a transparent bridge could (because of special hardware and architecture). Once a switch decides where the frame is sent, it passes the frame out the appropriate port (or ports). You can think of a switch as a device that creates instantaneous connections between various ports, on a frame by frame basis.

VLANs

Since the switch decides on a frame by frame basis which ports exchange data, it is a natural extension to put logic inside the switch to allow it to choose ports for special groupings. This grouping of ports is called a Virtual Local Area Network (VLAN). The switch makes sure that traffic from one group of ports never gets sent to other groups of ports (which would be routing). These port groups (VLANs) can each be considered an individual LAN segment.

VLANs are also described as broadcast domains. This is because of the transparent bridging algorithm, which says that broadcast packets (packets destined for the *all devices* address) be sent out all ports that are in the same group (that is, in the same VLAN). All ports that are in the same VLAN are also in the same broadcast domain.

Transparent Bridging Algorithm

The transparent bridging algorithm and spanning tree are covered in more detail elsewhere (Chapter 20: Troubleshoot Transparent Bridging Environments). When a switch receives a frame, it must decide what to do with that frame. It could ignore the frame; it could pass the frame out one

other port, or it could pass the frame out many other ports.

In order to know what to do with the frame, the switch learns the location of all devices on the segment. This location information is placed in a Content Addressable Memory table (CAM - named for the type of memory used to store these tables). The CAM table shows, for each device, the MAC address of the device, out which port that MAC address can be found, and with which VLAN this port is associated. The switch continually learns as frames are received into the switch. The CAM table of the switch is continually updated.

This information in the CAM table is used to decide how a received frame is handled. In order to decide where to send a frame, the switch looks at the destination MAC address in a received frame and looks up that destination MAC address in the CAM table. The CAM table shows which port the frame must be sent out in order for that frame to reach the specified destination MAC address. Here are the basic rules that a switch uses to carry out the frame forwarding responsibility:

If the destination MAC address is found in the CAM table, the switch sends the frame out the port that is associated with that destination MAC address in the CAM table. This is called *forwarding*.

If the associated port to send the frame out is the same port that the frame originally came in on, there is no need to send the frame back out that same port, and the frame is ignored. This is called *filtering*.

If the destination MAC address is not in the CAM table (the address is *unknown*), the switch sends the frame out all other ports that are in the same VLAN as the received frame. This is called flooding. It does not flood the frame out the same port on which the frame was received.

If the destination MAC address of the received frame is the broadcast address (FFFF.FFFF.FFFF), the frame is sent out all ports that are in the same VLAN as the received frame. This is also called flooding. The frame is not sent out the same port on which the frame was received.

Spanning Tree Protocol

As you have seen, the transparent bridging algorithm floods unknown and broadcast frames out of all the ports that are in the same VLAN as the received frame. This causes a potential problem. If the network devices that run this algorithm are connected together in a physical loop, flooded frames (like broadcasts) are passed from switch to switch, around and around the loop, forever. Dependent upon the physical connections involved, the frames can actually multiply exponentially due to the flooding algorithm, which can cause serious network problems.

There is a benefit to a physical loop in your network: it can provide redundancy. If one link fails, there is still another way for the traffic to reach its destination. In order to allow the benefits derived from redundancy, and not break the network because of flooding, a protocol called spanning tree was created. Spanning tree was standardized in the IEEE 802.1d specification.

The purpose of the spanning tree protocol (STP) is to identify and temporarily block the loops in a network segment or VLAN. The switches run the STP, and elect a root bridge or switch. The other

switches measure their distance from the root switch. If there is more than one way to get to the root switch, there is a loop. The switches trace the algorithm to determine which ports must be blocked in order to break the loop. STP is dynamic; if a link in the segment fails, ports that were originally blocking can possibly be changed to forwarding mode.

Trunking

Trunking is a mechanism that is most often used to allow multiple VLANs to function independently across multiple switches. Routers and servers can use trunking, as well, which allows them to live simultaneously on multiple VLANs. If your network only has one VLAN in it, you do not necessarily need trunking; but if your network has more than one VLAN, you probably want to take advantage of the benefits of trunking.

A port on a switch normally belongs to only one VLAN; any traffic received or sent on this port is assumed to belong to the configured VLAN. A trunk port, on the other hand, is a port that can be configured to send and receive traffic for many VLANs. It accomplishes this when it attaches VLAN information to each frame, a process called *tagging* the frame. Also, trunking must be active on both sides of the link; the other side must expect frames that include VLAN information for proper communication to occur.

There are different methods of trunking dependent upon the media that is used. Trunking methods for Fast Ethernet or Gigabit Ethernet are Inter-Switch Link (ISL) or 802.1q. Trunking over ATM uses LANE. Trunking over FDDI uses 802.10.

EtherChannel

EtherChannel is a technique that is used when you have multiple connections to the same device. Rather than each link function independently, EtherChannel groups the ports together to work as one unit. It distributes traffic across all the links and provides redundancy if one or more links fail. EtherChannel settings must be the same on both sides of the links involved in the channel. Normally, spanning tree would block all of these parallel connections between devices because they are loops, but EtherChannel runs *underneath* spanning tree, so that spanning tree thinks all the ports within a given EtherChannel are only a single port.

Multilayer Switching (MLS)

Multilayer switching (MLS) is the ability of a switch to forward frames based on information in the layer-three and sometimes layer-four header. This usually applies to IP packets but now also can occur for IPX packets. The switch learns how to handle these packets when it communicates with one or more routers. With a simplified explanation, the switch watches how the router processes a packet, and then the switch processes future packets in this same flow. Traditionally, switches have been much faster at switching frames than routers, so to have them offload traffic from the router can result in significant speed improvements. If something changes in the network, the router can tell the switch to erase its layer-three cache and build it from scratch again as the situation evolves. The protocol used to communicate with the routers is called Multilayer Switching Protocol (MLSP).

How to Learn About These Features

These are just some of the basic features that switches support. More are added every day. It is important to understand how your switches work, which features you use, and how those features need to work. One of the best places to learn this information about Cisco switches is on the Cisco web site. Go to and under the section *Service & Support*, choose *Technical Documents*. From

here, choose *Documentation Home Page* . Documentation sets for all Cisco products can be found here. The *Multilayer LAN Switches* link leads you to documentation for all Cisco LAN switches. In order to learn about the features of a switch, read the *Software Configuration Guide* for the particular release of software that you use. The software configuration guides give you background information about what the feature does and what commands to use to configure it on your switch. All this information is free on the web. You do not even need an account for this documentation; it is available to anyone. Some of these configuration guides can be read in an afternoon and are well worth the time spent.

Another part of the Cisco web site is populated by the Cisco Support and Documentation web site. It is filled with information designed to help you implement, maintain, and troubleshoot your network. Go to the [Support and Documentation](#) web site to get detailed support information by specific products or technologies.

Suggestion to Troubleshoot General Switch

There are many ways to troubleshoot a switch. As the features of switches grow, the possible things that can break also increase. To troubleshoot effectively, develop an approach or test plan rather than a hit-and-miss approach. Here are some general suggestions:

Take the time to become familiar with normal switch operation. The Cisco web site has a tremendous amount of technical information that describes how their switches work, as mentioned in the previous section. The configuration guides in particular are very helpful. Many cases are opened that are solved with information from the product configuration guides.

- For the more complex situations, have an accurate physical and logical map of your network. A physical map shows how the devices and cables are connected. A logical map shows what segments (VLANs) exist in your network and which routers provide routing services to these segments. A spanning tree map is highly useful to troubleshoot complex issues. Because of the ability of a switch to create different segments with the implementation of VLANs, the physical connections alone do not tell the whole story; one has to know how the switches are configured to determine which segments (VLANs) exist and to know how they are logically connected.

Have a plan. Some problems and solutions are obvious; some are not. The symptoms that you see in your network can be the result of problems in another area or layer. Before you jump to conclusions, try to verify in a structured way what works and what does not. Since networks can be complex, it is helpful to isolate possible problem domains. One way to do this is to use the OSI seven-layer model. For example: check the physical connections involved (layer 1); check connectivity issues within the VLAN (layer 2), and check connectivity issues across different VLANs (layer 3), and so on. If there is a correct configuration on the switch, many of the problems you encounter are related to physical layer issues (physical ports and cables). Today, switches are involved in layer-three and four issues, which incorporate intelligence to switch packets based on information derived from routers, or actually have routers that live inside the switch (layer-three or layer-four switching).

Do not assume a component works, you must check it first. This can save you a lot of wasted time. For example, if a PC is not able to log in to a server across your network, there are many things that can be wrong. Do not skip the basic things and assume that something works; someone can have changed something and not told you. It only takes a minute to check some

of the basic things (for example, that the ports involved are connected to the right place and are active), which could save you many wasted hours.

Troubleshoot Port Connectivity Problems

If the port does not work, nothing works! Ports are the foundation of your switching network. Some ports have special significance because of their location in the network and the amount of traffic they carry. These ports include connections to other switches, routers, and servers. These ports can be more complicated to troubleshoot because they often take advantage of special features like trunking and EtherChannel. The rest of the ports are significant, as well, because they connect the actual users of the network.

Many things can cause a port to be non-functional: hardware issues, configuration issues, and traffic issues. These categories are explored a little deeper.

Hardware Issues

General

Port functionality requires two active ports connected by a active cable (of the correct type). The default of most Cisco switches is to have a port in *notconnect* state, which means that it is currently not connected to anything, but it wants to connect. If you connect a good cable to two switch ports in the *notconnect* state, the link light becomes green for both ports, and the port status says *connected*, which means the port is up as far as layer one is concerned. These paragraphs point out items for which to check if layer one is not up.

Check the port status for both ports involved. Make sure that neither port involved in the link is shutdown. The administrator possibly can have shut down one or both ports. Software inside the switch can have shut the port down because of configuration error conditions. If one side is shut down and the other is not, the status on the enabled side is *notconnect* (because it does not sense a neighbor on the other side of the wire). The status on the shutdown side says something like *disable* or *errDisable* (dependent upon what actually shut the port down). The link does not come up unless both ports are enabled.

When you hook up a good cable (again, if it is of the correct type) between two enabled ports they show a green link light within a few seconds. Also, the port state shows *connected* in the command line interface (CLI). At this point, if you do not have link, your problem is limited to three things: the port on one side, the port on the other side, or the cable in the middle. In some cases, there are other devices involved: media converters (fiber to copper, and so on), or on Gigabit links you can have gigabit interface connectors (GBICs). Still, this is a reasonably limited area to search.

Media converters can add noise to a connection or weaken the signal if they do not function correctly. They also add extra connectors that can cause problems and are another component to debug.

Check for loose connections. Sometimes a cable appears to be seated in the jack, but it actually is not; unplug the cable and re-insert it. You must also look for dirt, lost, or broken pins. Do this for both ports involved in the connection.

The cable can be plugged in to the wrong port, which commonly happens. Make sure both ends of the cable are plugged in to the ports where you really want them.

You can have link on one side and not on the other. Check both sides for link. A single broken wire can cause this type of problem.

A link light does not guarantee that the cable is fully functional. It can have encountered physical stress that causes it to be functional at a marginal level. Usually, you notice this by the port that has lots of packet errors.

In order to determine if the cable is the problem, swap it with a known good cable. Do not just swap it with any other cable; make sure that you swap it with a cable that you know is good and is of the correct type.

If this is a very long cable run (underground, across a large campus, for example), it is nice to have a sophisticated cable tester. If you do not have a cable tester, you can consider these:

Try different ports to see if they come up with this long cable.

Connect the port in question to another port in the same switch just to see if the port links up locally.

Temporarily relocate the switches near each other, so you can try out a known good cable.

Copper

Make sure that you have the correct cable for the type of connection that you make. Category 3 cable can be used for 10MB UTP connections, but category 5 must be used for 10/100 connections.

A straight-through RJ-45 cable is used for end-stations, routers, or servers to connect to a switch or hub. An Ethernet crossover cable is used for switch to switch or hub to switch connections. This is the pin-out for an Ethernet crossover cable. Maximum distances for Ethernet or Fast Ethernet copper wires are 100 meters. A good general rule of thumb is that when you cross an OSI layer, as between a switch and a router, use a straight-through cable; when you connect two devices in the same OSI layer, as between two routers or two switches, use a cross over cable. For purposes of this rule only, treat a workstation like a router.

These two graphics show the pin-outs required for a switch-to-switch crossover cable.

Fiber

For fiber, make sure that you have the correct cable for the distances involved and the type of fiber ports that is used (single mode, multi-mode). Ensure the ports that are connected together are both single mode and both multi-mode ports. Single mode fiber generally reaches 10 kilometers, and multi-mode fiber can usually reach 2 kilometers, but there is the special case of 100BaseFX multi-mode used in half-duplex mode, which can only go 400 meters.

For fiber connections, make sure the transmit lead of one port is connected to the receive lead of the other port, and vice versa; transmit to transmit, receive to receive, does not work.

For gigabit connections, GBICs need to be matched on each side of the connection. There are different types of GBICs dependent upon the cable and distances involved: Short wavelength (SX), long wavelength/long haul (LX/LH), and extended distance (ZX).

An SX GBIC needs to connect with an SX GBIC; an SX GBIC does not link with an LX GBIC. Also, some gigabit connections require conditioning cables dependent upon the lengths involved. Refer to the GBIC installation notes.

If your gigabit link does not come up, check to make sure the flow control and port negotiation settings are consistent on both sides of the link. There can be incompatibilities in the implementation of these features if the switches that are connected are from different vendors. If in doubt, turn these features off on both switches.

Configuration Issues

Another cause of port connectivity issues is incorrect software configuration of the switch. If a port has a solid orange light, that means that software inside the switch shut down the port, either by way of the user interface or by internal processes.

Make sure that the administrator has not shut down the ports involved (as mentioned). The administrator can have manually shut down the port on one side of the link or the other. This link does not come up until you re-enable the port; check the port status.

Some switches, such as the Catalyst 4000/5000/6000, can shut down the port if software processes inside the switch detect an error. When you look at the port status, it reads *errDisable*. You must fix the configuration problem and then manually take the port out of *errDisable* state. Some newer software versions (CatOS 5.4(1) and later) have the ability to automatically re-enable a port after a configurable amount of time spent in the *errDisable* state. These are some of the causes for this *errDisable* state:

EtherChannel Misconfiguration: If one side is configured for EtherChannel and the other is not, it can cause the spanning tree process to shut down the port on the side configured for EtherChannel. If you try to configure EtherChannel but the ports involved do not have the same settings (speed, duplex, trunking mode, and so on) as their neighbor ports across the link, it could cause the *errDisable* state. It is best to set each side for the EtherChannel *desirable* mode if you want to use EtherChannel. Sections later on talk in depth about how to configure the EtherChannel.

Duplex Mismatch: If the switch port receives a lot of late collisions, this usually indicates a duplex mismatch problem. There are other causes for late collisions: a bad NIC, cable segments that are too long, but the most common reason today is a duplex mismatch. The full duplex side thinks it can send whenever it wants to. The half-duplex side only expects packets at certain times - not at "any" time.

BPDU Port-guard: Some newer versions of switch software can monitor if portfast is enabled on a port. A port that uses portfast must be connected to an end-station, not to devices that generate spanning tree packets called BPDUs. If the switch notices a BPDU that comes in a port that has portfast enabled, it puts the port in *errDisable* mode.

UDLD: Unidirectional Link Detection is a protocol on some new versions of software that discovers if communication over a link is one-way only. A broken fiber cable or other cable/port issues can cause this one-way only communication. These partially functional links can cause problems when the switches involved do not know that link is partially broken. Spanning tree loops can occur with this problem. UDLD can be configured to put a port in *errDisable* state when it detects a unidirectional link.

Native VLAN mismatch: Before a port has trunking turned on, it belongs to a single VLAN. When trunking is turned on, the port can carry traffic for many VLANs. The port still remembers the VLAN it was in before trunking was turned on, which is called the native VLAN. The native VLAN is central to 802.1q trunking. If the native VLAN on each end of the link does not match, a port goes into the `errDisable` state.

Other: Any process within the switch that recognizes a problem with the port can place it in the `errDisable` state.

Another cause of inactive ports is when the VLAN they belong to disappears. Each port in a switch belongs to a VLAN. If that VLAN is deleted, the port becomes inactive. Some switches show a steady orange light on each port where this has happened. If you come in to work one day and see hundreds of orange lights, do not panic; it could be that all the ports belonged to the same VLAN and someone accidentally deleted the VLAN that the ports belonged to. When you add the VLAN back into the VLAN table, the ports become active again. A port remembers its assigned VLAN.

If you have link and the ports show connected, but you cannot communicate with another device, this can be particularly perplexing. It usually indicates a problem higher than the physical layer: layer 2 or layer 3. Try these things.

Check the trunking mode on each side of the link. Make sure both sides are in the same mode. If you turn the trunking mode to "on" (as opposed to "auto" or "desirable") for one port, and the other port has the trunking

mode set to "off", they are not able to communicate. Trunking changes the format of the packet; the ports need to agree to what format they use on the link, or they do not understand each other.

Make sure all devices are in the same VLAN. If they are not in the same VLAN, a router must be configured to allow the devices to communicate.

Make sure your layer three addressing is correctly configured.

Traffic Issues

In this section describes some of the things you can learn when you look at that traffic information of a port. Most switches have some way to track the packets when they go in and out of a port. Commands that generate this type of output on the Catalyst 4000/5000/6000 switches are **show port** and **show mac**. Output from these commands on the 4000/5000/6000 switches is described in the switch command references.

Some of these port traffic fields show how much data is transmitted and received on the port. Other fields show how many error frames are encountered on the port. If you have a large amount of alignment errors, FCS errors, or late collisions, this can indicate a duplex mismatch on the wire. Other causes for these types of errors can be bad network interface cards or cable problems. If you have a large number of deferred frames, it is a sign that your segment has too much traffic; the switch is not able to send enough traffic on the wire to empty its buffers. Consider the removal

of some devices to another segment.

Switch Hardware Failure

If you have tried everything you can think of and the port does not work, there can be faulty hardware.

Sometimes ports are damaged by Electro-Static Discharge (ESD). You can or cannot see any indication of this.

Look at the power-on self-test (POST) results from the switch to see if there were any failures indicated for any part of the switch.

If you see behavior that can only be considered "strange," this could indicate hardware problems, but it could also indicate software problems. It is usually easier to reload the software than it is to get new hardware. Try to work with the switch software first.

The operating system can have a bug. If you load a newer operating system, it could fix this. You can research known bugs if you read the release notes for the version of code you use or use [Cisco Bug ToolKit](#).

The operating system could have somehow become corrupted. If you reload the same version of the operating system, you could fix the problem.

If the status light on the switch flashes orange, this usually means there is some kind of hardware problem with the port or the module or the switch. The same thing is true if the port or module status indicates *faulty*.

Before you exchange the switch hardware, you can try a few things:

Reseat the module in the switch. If you do this with the power on, make sure the module is hot swappable. If in doubt, turn the switch off before you reseat the module or refer to the hardware installation guide. If the port is built into the switch, ignore this step.

Reboot the switch. Sometimes this causes the problem to disappear; this is a workaround, not a fix.

Check the switch software. If this is a new installation, remember that some components can only work with certain releases of software. Check the release notes or the hardware installation and configuration guide for the component you install.

If you are reasonably certain that you have a hardware problem, replace the faulty component.

Troubleshoot Ethernet 10/100Mb Half/Full Duplex Auto-Negotiation

Objectives

This section presents general information used to troubleshoot and a discussion of techniques to troubleshoot Ethernet auto-negotiation.

This section shows how to determine the current behavior of a link. It goes on to show how users can control the behavior, as well as explain situations when auto-negotiation fails.

Many different Cisco Catalyst Switches and Cisco Routers support auto-negotiation. This section focuses on auto-negotiation between Catalyst 5000 Switches. The concepts explained here can also be applied to the other types of devices.

Introduction

Auto-negotiation is an optional function of the IEEE 802.3u Fast Ethernet standard that enables devices to automatically exchange information over a link about speed and duplex abilities.

Auto-negotiation is targeted at ports, which are allocated to areas where transient users or devices connect to a network. For example, many companies provide shared offices or cubes for Account Managers and System Engineers to use when they are in the office rather than on the road. Each office or cube has an Ethernet port permanently connected to the network of the office. Because it is not possible to ensure that every user has either a 10Mb, a 100Mb Ethernet, or a 10/100Mb card in their laptop, the switch ports that handle these connections must be able to negotiate their speed and duplex mode. The alternative is able to provide both a 10Mb and a 100Mb port in each office or cube and label them accordingly.

Auto-negotiation must not be used for ports that support network infrastructure devices, such as switches and routers or other non-transient end systems such as servers and printers. Although auto-negotiation for speed and duplex is normally the default behavior on switch ports that are capable of it, ports connected to fixed devices must always be configured for the correct behavior rather than allowed to negotiate it. This eliminates any potential negotiation issues and ensures that you always know exactly how the ports need to operate. For example, a 10/100BaseTX Ethernet switch-to-switch link that has been configured for 100Mb Full Duplex only operates at that speed and mode. There is no possibility for the ports to downgrade the link to a slower speed within a port reset or a switch reset. In the event that the ports cannot operate as configured, they must not pass any traffic. On the other hand, a switch-to-switch link that has been allowed to negotiate its behavior can operate at 10Mb half-duplex. A non-functional link is usually easier to discover than a link, which is operational, but does not operate at the expected speed or mode.

One of the most common causes of performance issues on 10/100Mb Ethernet links is when one port on the link operates at half-duplex, while the other port operates at full duplex. This occasionally happens when one or both ports on a link are reset and the auto-negotiation process does not result in both link partners that have the same configuration. It also happens when users reconfigure one side of a link and forget to reconfigure the other side. Many performance-related support calls are avoided if you create a policy that requires ports for all non-transient devices to be configured for their required behavior and enforce the policy with adequate change control measures.

Troubleshoot Ethernet Auto-negotiation Between Network Infrastructure Devices

Procedures and/or Scenarios

Table 22-2: Auto-Negotiation Connectivity Issues

Possible Problem	Solution
Was the current behavior of the link auto negotiated?	1. Use the show port mod_num/port_num command to determine the current behavior of the link. If both link partners (interfaces at either end of the link) indicate have an "a-" prefix on their Duplex and Speed status fields, auto-negotiation was probably successful.
auto-negotiation not supported.	2. Issue the show port capabilities mod_num/port_num command to verify that your modules support auto-negotiation.
auto-negotiation do not work on Catalyst switches.	3. Use the set port speed mod_num/port_num auto command on a Catalyst to configure auto-negotiation. 4. Try different ports or modules. 5. Try to reset the ports. 6. Try different patch cables. 7. Turn the devices off and back on again.
auto-negotiation do not work on Cisco routers.	8. Issue the correct Cisco IOS command to enable auto-negotiation (if available) 9. Try different interfaces. 10. Try to reset the interfaces. 11. Try different patch cables. 12. Turn the devices off and back on again.

Configure and Troubleshoot Ethernet 10/100Mb Auto-Negotiation Example

This section goes through an examination of the behavior of a 10/100Mb Ethernet port that supports auto-negotiation. It also shows how to make changes to its default behavior and how to restore it to the default behavior.

Tasks to Perform

Examine the capabilities of the ports.

Configure auto-negotiation for port 1/1 on both switches.

Determine if the speed and duplex mode are set to auto-negotiate.

Change the speed on port 1/1 in Switch A to 10Mb.

Understand the meaning of the "a-" prefix on the duplex and speed status fields.

View the duplex status of port 1/1 on Switch B.

Understand the Duplex mismatch error.

Understand the Spanning Tree error messages.

Change the duplex mode to half on port 1/1 on Switch A.

Set the duplex mode and speed of port 1/1 on Switch B.

Restore the default duplex mode and speed to ports 1/1 on both switches.

View the changes of the port status on both switches.

Step-by-Step

Perform these steps:

The **show port capabilities 1/1** command displays the capabilities of an Ethernet 10/100BaseTX 1/1 port on Switch A.

Enter this command for both of the ports you troubleshoot. Both ports must support the speed and duplex capabilities shown if they are supposed to use auto-negotiation.

```
Switch-A> (enable) show port capabilities 1/1
Model WS-X5530
Port 1/1
Type 10/100BaseTX
Speed auto,10,100
Duplex half, full
```

Auto-negotiation is configured for both speed and duplex mode on port 1/1 of both switches if you enter the **set port speed 1/1 auto** command (auto is the default for ports that support auto-negotiation).

```
Switch-A> (enable) set port speed 1/1 auto
Port(s) 1/1 speed set to auto detect.
Switch-A (enable)
```

Note: The **set port speed {mod_num/port_num} auto** command also sets the duplex mode to auto. There is no **set port duplex {mod_num/port_num} auto** command.

The **show port 1/1** command displays the status of ports 1/1 on Switches A and B.

```
Switch-A> (enable) show port 1/1
Port Name          Status      Vlan      Level Duplex Speed Type
-----
1/1                connected  1         normal a-full a-100 10/100BaseTX

Switch-B> (enable) show port 1/1
```

Port	Name	Status	Vlan	Level	Duplex	Speed	Type
1/1		connected	1	normal	a-full	a-100	10/100BaseTX

Note that most of the normal output from the **show port {mod_num/port_num}** command has been omitted.

The "a-" prefixes on the "full" and "100" indicate that this port has not been hard coded (configured) for a specific duplex mode or speed. Therefore, it can auto-negotiate its duplex mode and speed if the device it is connected to (its Link Partner) also can auto-negotiate its duplex mode and speed. Also note that the status is "connected" on both ports, which means that a link pulse has been detected from the other port. The status can be "connected" even if duplex has been incorrectly negotiated or misconfigured.

In order to demonstrate what happens when one link partner auto-negotiates and the other Link Partner does not, the speed on port 1/1 in Switch A is set to 10Mb with the **set port speed 1/1 10** command.

```
Switch-A> (enable) set port speed 1/1 10
Port(s) 1/1 speed set to 10Mbps.
Switch-A> (enable)
```

Note: If you hard code the speed on a port, it disables all auto-negotiation functionality on the port for speed and duplex.

When a port has been configured for a speed, its duplex mode is automatically configured for the previously negotiated mode. In this case, full duplex. When you enter the **set port speed 1/1 10** command caused the duplex mode on port 1/1 to be configured as if the command **set port duplex 1/1 full** had also been entered. This is explained next.

Understand the meaning of the "a-" prefix in the Duplex and Speed status fields.

The absence of the "a-" prefix in the status fields of the output from the **show port 1/1** command on Switch A shows that the duplex mode is now configured for "full," and the speed is now configured for "10."

```
Switch-A> (enable) show port 1/1
Port Name      Status      Vlan      Level Duplex Speed Type
-----
1/1           connected  1         normal full   10   10/100BaseTX
```

The **show port 1/1** command on Switch B indicates that the port now operates at half-duplex and 10Mb.

```
Switch-B> (enable) show port 1/1
Port Name      Status      Vlan      Level Duplex Speed Type
```

```
-----  
1/1          connected 1          normal a-half a-10 10/100BaseTX
```

This step shows that it is possible for a Link Partner to detect the speed at which the other Link Partner operates, even though the other Link Partner is not configured for auto-negotiation. Sensing the type of electrical signal that arrives to discover if it is 10Mb or 100Mb does this. This is how Switch B determined that port 1/1 must operate at 10Mb.

It is not possible to detect the correct duplex mode in the same way that the correct speed can be detected. In this case, where the 1/1 port of Switch B is configured for auto-negotiation and the port of Switch A is not, the 1/1 port of Switch B was forced to select the default duplex mode. On Catalyst Ethernet ports, the default mode is auto-negotiate, and if auto-negotiation fails, then half-duplex.

This example also shows that a link can be successfully connected when there is a mismatch in the duplex modes. Port 1/1 on Switch A is configured for full duplex while port 1/1 on Switch B has defaulted to half-duplex. In order to avoid this, always configure both Link Partners.

The "a-" prefix on the Duplex and Speed status fields does not always mean the current behavior was negotiated. Sometimes it only means that the port has not been configured for a speed or duplex mode. The previous output from Switch B shows Duplex as "a-half" and Speed as "a-10" which indicates that the port operates at 10Mb in half-duplex mode. In this example, the link partner on this port (port 1/1 on Switch A) is configured for "full" and "10Mb." It was not possible for port 1/1 on Switch B to have auto-negotiated its current behavior. This proves that the "a-" prefix only indicates a willingness to perform auto-negotiation but not that auto-negotiation actually took place.

Understand the Duplex Mismatch error message.

This message about a duplex mode mismatch is displayed on Switch A after the speed on port 1/1 was changed to 10Mb. The mismatch was caused by the 1/1 port of Switch B, which default to half-duplex because it sensed its Link Partner could no longer perform auto-negotiation.

```
%CDP-4-DUPLEXMISMATCH:Full/half-duplex mismatch detected on
```

It is important to note that this message is created the Cisco Discovery Protocol (CDP), not the 802.3 auto-negotiation protocol. CDP can report problems it discovers, but it typically does not automatically fix them. A duplex mismatch can or cannot result in an error message. Another indication of a duplex mismatch are the rapidly increased FCS and alignment errors on the half-duplex side and "runts" on the full duplex port (as seen in a **sh port {mod_num/port_num}**).

Understand the Spanning Tree messages.

In addition to the duplex mismatch error message, you can also see these Spanning Tree messages when you change the speed on a link. A discussion of Spanning Tree is beyond the scope of this document; refer the chapter on Spanning Tree for more information on Spanning Tree.

```
%PAGP-5-PORTFROMSTP:Port 1/1 left bridge port 1/1
%PAGP-5-PORTTOSTP:Port 1/1 joined bridge port 1/1
```

In order to demonstrate what happens when the duplex mode has been configured, the mode on port 1/1 in Switch A is set to half with the **set port duplex 1/1 half** command.

```
Switch-A> (enable) set port duplex 1/1 half
Port(s) 1/1 set to half-duplex.
Switch-A> (enable)
```

The **show port 1/1** command shows the change in the Duplex mode on this port.

```
Switch-A> (enable) sh port 1/1
Port Name          Status      Vlan      Level Duplex Speed Type
-----
1/1                connected  1         normal half  10    10/100BaseTX
```

At this point, ports 1/1 on both switches operate at half-duplex. Port 1/1 on Switch B is still configured to auto negotiate, as shown in this output of the **show port 1/1** command.

```
Switch-B> (enable) show port 1/1
Port Name          Status      Vlan      Level Duplex Speed Type
-----
1/1                connected  1         normal a-half a-10  10/100BaseTX
```

This step shows how to configure the duplex mode on port 1/1 in Switch B to half. This is consistent with the recommended policy to configure both link partners in the same way.

In order to implement the policy to ways configure both link partners for the same behavior, this step now sets the duplex mode to half and speed to 10 on port 1/1 in Switch B.

Here is the output when you enter the **set port duplex 1/1 half** command on Switch B:

```
Switch-B> (enable) set port duplex 1/1 half
Port 1/1 is in auto-sensing mode.
Switch-B> (enable)
```

The **set port duplex 1/1 half** command failed because this command is not valid if auto-negotiation is enabled. This also means that this command does not disable auto-negotiation. Auto-negotiation can only be disabled with the **set port speed {mod_num/port_num {10 | 100} }** command.

Here is the output when you enter the **set port speed 1/1 10** command on Switch B:

```
Switch-B> (enable) set port speed 1/1 10
Port(s) 1/1 speed set to 10Mbps.
Switch-B> (enable)
```

Now the **set port duplex 1/1 half** command on Switch B works:

```
Switch-A> (enable) set port duplex 1/1 half
Port(s) 1/1 set to half-duplex.
Switch-A> (enable)
```

The **show port 1/1** command on Switch B shows that the ports is now configured for half-duplex and 10Mb.

```
Switch-B> (enable) show port 1/1
Port  Name          Status      Vlan      Level Duplex Speed Type
-----
1/1          connected  1         normal half   10   10/100BaseTX
```

Note: The **set port duplex {mod_num/port_num {half | full }}** command is dependent on the **set port speed {mod_num/port_num {10 | 100 }}** command. In other words, you must set the speed before you can set the duplex mode.

Configure ports 1/1 on both switches to auto negotiate with the **set port speed 1/1 auto** command.

```
Switch-A> (enable) set port speed 1/1 auto
Port(s) 1/1 speed set to auto detect.
Switch-A> (enable)
```

Note: Once a duplex mode of a port has been configured to something other than auto, the only way to configure the port to auto sense its duplex mode is to issue the **set port speed {mod_num/port_num} auto** command. There is no **set port duplex {mod_num/port_num} auto** command. In other words, if you issue the **set port speed {mod_num/port_num} auto** command, it resets both port speed detection and duplex mode detection to auto.

Examine the status of ports 1/1 on both switches with the **show port 1/1** command.

```
Switch-A> (enable) show port 1/1
Port  Name          Status      Vlan      Level Duplex Speed Type
-----
1/1          connected  1         normal a-full a-100 10/100BaseTX
Switch-B> (enable) show port 1/1
```

Port	Name	Status	Vlan	Level	Duplex	Speed	Type
1/1		connected	1	normal	a-full	a-100	10/100BaseTX

Both ports are now set to their default behavior of auto-negotiation. Both ports have negotiated full duplex and 100Mb.

Before You Call the Cisco Systems Technical Support Team

Before you call the Cisco Systems Technical Support Website, ensure you have read through this article and completed the actions suggested for your system problems. Additionally, document the results so that Cisco can better assist you:

Capture the output of **show version** from all of the affected devices.

Capture the output of **show port mod_num/port_num** from all of the affected ports.

Capture the output of **show port mod_num/port_num capabilities** from all of the affected ports.

Configure EtherChannel Switch-to-Switch Connections on Catalyst 4000/5000/6000 Switches

EtherChannel allows multiple physical Fast Ethernet or Gigabit Ethernet links to be combined into one logical channel. This allows traffic among the links to load share in the channel, as well as redundancy in the event that one or more links in the channel fail. EtherChannel can be used to interconnect LAN switches, routers, servers, and clients through unshielded twisted-pair (UTP) wiring or single mode and multi-mode fiber.

EtherChannel is an easy way to aggregate bandwidth between critical networking devices. On the Catalyst 5000, a channel can be created from two ports that make it a 200Mbps link (400Mbps full-duplex) or four ports that make it a 400Mbps link (800Mbps full-duplex). Some cards and platforms also support Gigabit EtherChannel and have the ability to use from two to eight ports in an EtherChannel. The concept is the same no matter what speeds or number of links are involved. Normally the spanning tree protocol (STP) considers these redundant links between two devices to be loops and sets the redundant links to the blocking mode. This effectively makes these links inactive (that provide only backup capabilities if the main link fails). When you use Cisco IOS 3.1.1 or greater, spanning tree treats the channel as one big link, so all the ports in the channel can be active at the same time.

This section takes you through the steps to configure EtherChannel between two Catalyst 5000 switches and show you the results of the commands as they are executed. Catalyst 4000 and 6000 switches could have been used in the scenarios presented in this document to obtain the same results. For the Catalyst 2900XL and 1900/2820, the command syntax is different, but the EtherChannel concepts are the same.

EtherChannel can be configured manually if you type in the appropriate commands, or it can be configured automatically if the switch negotiates the channel with the other side with the Port

Aggregation Protocol (PAgP). It is recommended to use PAgP desirable mode to configure EtherChannel whenever possible since manual configuration of EtherChannel can create some complications. This document gives examples of how to configure EtherChannel manually and examples of how to configure EtherChannel with PAgP. Also included is how to troubleshoot EtherChannel and how to use trunking with EtherChannel. In this document, the terms EtherChannel, Fast EtherChannel, Gigabit EtherChannel or channel all refer to EtherChannel.

Contents

[Tasks for Manual Configuration of EtherChannel](#)

[Verify the EtherChannel Configuration](#)

[Use PAgP to Automatically Configure EtherChannel \(preferred method\)](#)

[Trunking and EtherChannel](#)

[Troubleshooting EtherChannel](#)

[Commands Used in this Document](#)

This figure illustrates this test environment. The configuration of the switches has been cleared with the **clear config all** command. Then, the prompt was changed with **set system name**. An IP address and mask were assigned to the switch for management purposes with **set int sc0 172.16.84.6 255.255.255.0** for SwitchA and **set int sc0 172.16.84.17 255.255.255.0** for SwitchB. A default gateway was assigned to both switches with **set ip route default 172.16.84.1**.

The switch configurations were cleared so that they start from the default conditions. The switches were given names to identify them from the prompt on the command line. The IP addresses were assigned so that you could ping between the switches to test them. The default gateway was not used.

Many of the commands display more output than is needed. Extraneous output is deleted from this document.

Tasks for Manual Configuration of EtherChannel

This is a synopsis of directions to manually configure the EtherChannel:

[Show the Cisco IOS version and modules used in this document.](#)

[Verify that EtherChannel is supported on the ports.](#)

[Verify that the ports are connected and operational.](#)

[Verify that the ports to be grouped have the same settings.](#)

[Identify Valid Port Groups.](#)

[Create the channel.](#)

Step-by-Step

These are the steps to manually configure the EtherChannel.

The **show version** command displays the software version the switch runs. The **show module** command lists which modules are installed in the switch.

```
Switch-A show version
WS-C5505 Software, Version MpsSW: 4.5(1) NmpSW: 4.5(1)
Copyright (c) 1995-1999 by Cisco Systems
?

Switch-A show module
Mod Module-Name          Ports Module-Type          Model      Serial-Num Status
-----
1                0      Supervisor III          WS-X5530  006841805 ok
2                24     10/100BaseTX Ethernet WS-X5225R 012785227 ok
?
```

Verify that EtherChannel is supported on the ports, **show port capabilities** appears in versions 4.x and greater. If you have an earlier Cisco IOS than 4.x, you must skip this step. Not every Fast Ethernet module supports EtherChannel. Some of the original EtherChannel modules have "Fast EtherChannel" written on the bottom left corner of the module (as you face it in the switch) which tells you that the feature is supported. This convention was abandoned on later modules. The modules in this test do not say "Fast EtherChannel" on them, but they do support the feature.

```
Switch-A show port capabilities
Model                WS-X5225R
Port                 2/1
Type                 10/100BaseTX
Speed                auto,10,100
Duplex               half,full
Trunk encap type     802.1Q,ISL
Trunk mode           on,off,desirable,auto,nonegotiate
Channel              2/1-2,2/1-4
Broadcast suppression percentage(0-100)
Flow control         receive-(off,on),send-(off,on)
Security             yes
Membership           static,dynamic
```

```

Fast start          yes
Rewrite            yes
Switch-B show port capabilities
Model              WS-X5234
Port               2/1
Type               10/100BaseTX
Speed              auto,10,100
Duplex             half,full
Trunk encap type   802.1Q,ISL
Trunk mode         on,off,desirable,auto,nonegotiate
Channel            2/1-2,2/1-4
Broadcast suppression percentage(0-100)
Flow control       receive-(off,on),send-(off,on)
Security           yes
Membership         static,dynamic
Fast start         yes
Rewrite            no

```

A port that does not support EtherChannel looks like this:

```

Switch show port capabilities
Model              WS-X5213A
Port               2/1
Type               10/100BaseTX
Speed              10,100,auto
Duplex             half,full
Trunk encap type   ISL
Trunk mode         on,off,desirable,auto,nonegotiate
Channel            no
Broadcast suppression pps(0-150000)
Flow control       no
Security           yes
Membership         static,dynamic
Fast start         yes

```

Verify that the ports are connected and operational. Before you connect the cables, this is the port status.

```

Switch-A show port
Port Name          Status      Vlan      Level Duplex Speed Type
-----
2/1                notconnect 1          normal  auto  auto 10/100BaseTX
2/2                notconnect 1          normal  auto  auto 10/100BaseTX
2/3                notconnect 1          normal  auto  auto 10/100BaseTX
2/4                notconnect 1          normal  auto  auto 10/100BaseTX

```

After you connect the cables between the two switches, this is the status.

```
1999 Dec 14 20:32:44 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1
1999 Dec 14 20:32:44 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/2
1999 Dec 14 20:32:44 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/3
1999 Dec 14 20:32:44 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/4
```

```
Switch-A show port
```

Port	Name	Status	Vlan	Level	Duplex	Speed	Type
2/1		connected	1	normal	a-full	a-100	10/100BaseTX
2/2		connected	1	normal	a-full	a-100	10/100BaseTX
2/3		connected	1	normal	a-full	a-100	10/100BaseTX
2/4		connected	1	normal	a-full	a-100	10/100BaseTX

```
Switch-B show port
```

Port	Name	Status	Vlan	Level	Duplex	Speed	Type
2/1		connected	1	normal	a-full	a-100	10/100BaseTX
2/2		connected	1	normal	a-full	a-100	10/100BaseTX
2/3		connected	1	normal	a-full	a-100	10/100BaseTX
2/4		connected	1	normal	a-full	a-100	10/100BaseTX

Since the switch configurations were cleared before this test started, the ports are in their default conditions. They are all in vlan1, and their speed and duplex are set to auto. After the connection of the cables, they negotiate to a speed of 100Mbps and full duplex. The status is connected, so you are able to ping the other switch.

```
Switch-A ping 172.16.84.17
172.16.84.17 is alive
```

In your network, you can set the speeds manually to 100Mbps and full duplex instead of reliance on auto-negotiation since you probably want your ports to always run at the fastest speed. For a discussion of auto-negotiation, see the section [Troubleshooting Ethernet 10/100Mb Half/Half/Full Duplex Auto-Negotiation](#).

Verify that the ports to be grouped have the same settings. This is an important point that is covered in more detail in the troubleshoot section. If the command to setup EtherChannel does not work, it is usually because the ports involved in the channel have configurations that differ from each other. This includes the ports on the other side of the link, as well as the local ports. In this case, since the switch configurations were cleared before this test started, the ports are in their default conditions. They are all in vlan1; their speed and duplex are set to auto, and all spanning tree parameters for each port are set the same. You saw from the output that after the cables are connected, the ports negotiate to a speed of 100Mbps and full duplex. Since spanning tree runs for each VLAN, it is easier to just configure the channel and respond to error messages than to try and check every spanning tree field for consistency for each port and VLAN in the channel.

Identify valid port groups. On the Catalyst 5000, only certain ports can be put together into a channel. These restrictive dependencies do not apply to all platforms. The ports in a channel on a Catalyst 5000 must be contiguous. Notice from the **show port capabilities** command

that for port 2/1, these are the possible combinations:

```
Switch-A show port capabilities
Model                WS-X5225R
Port                 2/1
Channel              2/1-2,2/1-4
```

Notice that this port can be a part of a group of two (2/1-2) or part of a group of four (2/1-4). There is something called an Ethernet Bundling Controller (EBC) on the module that causes these configuration limitations. Look at another port.

```
Switch-A show port capabilities 2/3
Model                WS-X5225R
Port                 2/3
Channel              2/3-4,2/1-4
```

This port can be grouped into a group of two ports (2/3-4) or into a group of four (2/1-4).

Note: Dependent upon the hardware, there can be additional restrictions. On certain modules (WS-X5201 and WS-X5203), you cannot form an EtherChannel with the last two ports in a "port group" unless the first two ports in the group already form an EtherChannel. A "port group" is a group of ports that is allowed to form an EtherChannel (2/1-4 is a port group in this example). For example, if you create separate EtherChannels with only two ports in a channel, you cannot assign ports 2/3-4 to a channel until you have first configured ports 2/1-2 to a channel, for the modules that have this restriction! Likewise, before you configure ports 2/6-7, you must configure ports 2/5-6. This restriction does not occur on the modules used for this document (WS-X5225R, WS-X5234).

Since you configure a group of four ports (2/1-4), this is within the approved grouping. you cannot assign a group of four to ports 2/3-6. This is a group of contiguous ports, but they do not start on the approved boundary, as shown by the **show port capabilities** command (valid groups would be ports 1-4, 5-8, 9-12, 13-16, 17-20, 21-24).

Create the channel. In order to create the channel, use the command **set port channel <mod/port on** for each switch. you recommend that you turn the ports off on one side of the channel or the other side with the **set port disable** command before you turn EtherChannel on manually. This avoids possible problems with spanning tree within the configuration process. Spanning tree can shut down some ports (with a port status of "errdisable") if one side is configured as a channel before the other side can be configured as a channel. Because of this possibility, it is much easier to create EtherChannels with PAgP, which is explained later in this document. In order to avoid this situation when you configure EtherChannel manually, you disable the ports on SwitchA, configure the channel on SwitchA, configure the channel on SwitchB, and then re-enable the ports on SwitchA.

First, verify that channeling is *off*.

```
Switch-A (enable) show port channel
No ports channelling
Switch-B (enable) show port channel
```

No ports channelling

Now disable the ports on SwitchA until both switches have been configured for EtherChannel so that spanning tree does not generate errors and shut down the ports.

```
Switch-A (enable) set port disable 2/1-4
Ports 2/1-4 disabled.
[output from SwitchA upon disabling ports]
1999 Dec 15 00:06:40 %PAGP-5-PORTFROMSTP:Port 2/1 left bridg1
1999 Dec 15 00:06:40 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
1999 Dec 15 00:06:40 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
1999 Dec 15 00:06:40 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
```

Turn the channel mode *toon* for SwitchA.

```
Switch-A (enable) set port channel 2/1-4 on
Port(s) 2/1-4 channel mode set to on.
```

Check the status of the channel. Notice that the channel mode has been set *toon*, but the status of the ports is disabled (because you disabled them earlier). The channel is not operational at this point, but it becomes operational when the ports are enabled.

```
Switch-A (enable) show port channel
Port  Status      Channel  Channel  Neighbor  Neighbor
      status      mode     status   device    port
-----
 2/1  disabled    on       channel
 2/2  disabled    on       channel
 2/3  disabled    on       channel
 2/4  disabled    on       channel
-----
```

Because SwitchA ports were (temporarily) disabled, SwitchB ports no longer have a connection. This message is displayed on the console of SwitchB when the ports of SwitchA were disabled.

```
Switch-B (enable)
2000 Jan 13 22:30:03 %PAGP-5-PORTFROMSTP:Port 2/1 left bridge port 2/1
2000 Jan 13 22:30:04 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
2000 Jan 13 22:30:04 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
2000 Jan 13 22:30:04 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
```

Turn on the channel for Switch B.

```
Switch-B (enable) set port channel 2/1-4 on
```

```
Port(s) 2/1-4 channel mode set to on.
```

Verify that channel mode is on for SwitchB.

```
Switch-B (enable) show port channel
Port  Status      Channel  Channel  Neighbor  Neighbor
      mode        status   device   port
-----
2/1   notconnect on       channel
2/2   notconnect on       channel
2/3   notconnect on       channel
2/4   notconnect on       channel
-----
```

Notice that the channel mode for SwitchB is on, but the status of the ports is *notconnect*. That is because SwitchA ports are still disabled.

Finally, the last step is to enable the ports on SwitchA.

```
Switch-A (enable) set port enable 2/1-4
Ports 2/1-4 enabled.
1999 Dec 15 00:08:40 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1-4
1999 Dec 15 00:08:40 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/1-4
1999 Dec 15 00:08:40 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/1-4
1999 Dec 15 00:08:40 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/1-4
```

Verify the Configuration

In order to verify that the channel is setup properly, do the **show port channel** command.

```
Switch-A (enable) show port channel
Port  Status      Channel  Channel  Neighbor  Neighbor
      mode        status   device   port
-----
2/1   connected on       channel  WS-C5505  066509957(Sw 2/1
2/2   connected on       channel  WS-C5505  066509957(Sw 2/2
2/3   connected on       channel  WS-C5505  066509957(Sw 2/3
2/4   connected on       channel  WS-C5505  066509957(Sw 2/4
-----
```

```
Switch-B (enable) show port channel
Port  Status      Channel  Channel  Neighbor  Neighbor
      mode        status   device   port
-----
2/1   connected on       channel  WS-C5505  066507453(Sw 2/1
2/2   connected on       channel  WS-C5505  066507453(Sw 2/2
2/3   connected on       channel  WS-C5505  066507453(Sw 2/3
-----
```

```
2/4 connected on channel WS-C5505 066507453(Sw 2/4
```

Spanning tree is shown to treat the ports as one logical port in this command. When the port is listed as 2/1-4, spanning tree handles ports 2/1, 2/2, 2/3 and 2/4 as one port.

```
Switch-A (enable) show spantree
```

```
VLAN 1
```

```
Spanning tree enabled
```

```
Spanning tree type ieee
```

```
Designated Root 00-10-0d-b2-8c-00
```

```
Designated Root Priority 32768
```

```
Designated Root Cost 8
```

```
Designated Root Port 2/1-4
```

```
Root Max Age 20 sec Hello Time 2 sec Forward Delay 15 sec
```

```
Bridge ID MAC ADDR 00-90-92-b0-84-00
```

```
Bridge ID Priority 32768
```

```
Bridge Max Age 20 sec Hello Time 2 sec Forward Delay 15 sec
```

Port	Vlan	Port-State	Cost	Priority	Fast-Start	Group-Method
2/1-4	1	forwarding	8	32	disabled	channel

EtherChannel can be implemented with different ways of traffic distribution across the ports in a channel. The EtherChannel specification does not dictate how the traffic must be distributed across the links in a channel. The Catalyst 5000 uses the last bit or the last two bits (dependent upon how many links are in the channel) of the source and destination mac addresses in the frame to determine which port in the channel to use. You see similar amounts of traffic on each of the ports in the channel if that traffic is generated by a normal distribution of MAC addresses on one side of the channel or the other. In order to verify that traffic goes over all the ports in the channel, you can use the **show mac** command. If your ports were active before you configured EtherChannel, you can reset the traffic counters to zero by the **clear counters** command, and then the traffic values represent how EtherChannel has distributed the traffic.

In this test environment, you did not get a real-world distribution because there are no workstations, servers, or routers that generate traffic. The only devices that generate traffic are the switches themselves. you issued some pings from SwitchA to SwitchB, and you can tell that the unicast traffic uses the first port in the channel. The Receive information in this case (Rcv-Unicast) shows how SwitchB distributed the traffic across the channel to SwitchA. A little lower in the output, the Transmit information (Xmit-Unicast) shows how SwitchA distributed the traffic across the channel to SwitchB. you also see that a small amount of switch-generated multicast traffic (Dynamic ISL, CDP) go out all four ports. The broadcast packets are ARP queries (for the default gateway - which does not exist here). If you had workstations that send packets through the switch to a destination on the other side of the channel, you would expect to see traffic that goes over each of the four links in the channel. You can monitor the packet distribution in your own network with the **show mac** command.

```
Switch-A (enable) clear counters
```

This command will reset all MAC and port counters reported in CLI and SNMP.

Do you want to continue (y/n) [n]? y

MAC and Port counters cleared.

Switch-A (enable) **show mac**

Port	Rcv-Unicast	Rcv-Multicast	Rcv-Broadcast
2/1	9	320	183
2/2	0	51	0
2/3	0	47	0
2/4	0	47	0
(...)			

Port	Xmit-Unicast	Xmit-Multicast	Xmit-Broadcast
2/1	8	47	184
2/2	0	47	0
2/3	0	47	0
2/4	0	47	0
(...)			

Port	Rcv-Octet	Xmit-Octet
2/1	35176	17443
2/2	5304	4851
2/3	5048	4851
2/4	5048	4851
(...)		

Last-Time-Cleared

Wed Dec 15 1999, 01:05:33

Use PAgP to Configure EtherChannel (Preferred Method)

The Port Aggregation Protocol (PAgP) facilitates the automatic creation of EtherChannel links with the exchange of packets between channel-capable ports. The protocol learns the capabilities of port groups dynamically and informs the nearby ports.

Once PAgP identifies correctly paired channel-capable links, it groups the ports into a channel. The channel is then added to the spanning tree as a single bridge port. A given outbound broadcast or multicast packet is transmitted out one port in the channel only, not out every port in the channel. In addition, outbound broadcast and multicast packets transmitted on one port in a channel are blocked from their return on any other port of the channel.

There are four user-configurable channel modes: on, off, auto, and desirable. PAgP packets are exchanged only between ports in **auto** and **desirable** mode. Ports configured in **on** or **off** mode do not exchange PAgP packets. The recommended settings for switches that you want to form an EtherChannel is to have both switches set to **desirable** mode. This gives the most robust behavior when one side or the other encounter error situations or be reset. The default mode of the channel is **auto**.

Both the **auto** and **desirable** modes allow ports to negotiate with connected ports to determine if they can form a channel based on criteria such as port speed, trunking state, native VLAN, and so on.

Ports can form an EtherChannel when they are in different channel modes as long as the modes are compatible:

A port in **desirable** mode can form an EtherChannel successfully with another port that is in **desirable** or **auto** mode.

A port in **auto** mode can form an EtherChannel with another port in **desirable** mode.

A port in **auto** mode cannot form an EtherChannel with another port that is also in **auto** mode since neither port initiates negotiation.

A port in **on** mode can form a channel only with a port in **on** mode because ports in **on** mode do not exchange PAgP packets.

A port in **off** mode does not form a channel with any port.

When you use EtherChannel, if a "SPANTREE-2: Channel misconfig - x/x-x will be disabled" or similar syslog message is displayed, it indicates a mismatch of EtherChannel modes on the connected ports. you recommend that you correct the configuration and re-enable the ports with the **set port enable** command. Valid EtherChannel configurations include these:

Table 22-5: Valid EtherChannel Configurations

Port Channel Mode	Valid Neighbor Port Channel Mode(s)
desirable	desirable or auto
auto (default)	desirable or auto ¹
on	on
off	off

¹If both the local and neighbor ports are in **auto** mode, an EtherChannel bundle does not form.

Here is a summary of all the possible channeling mode scenarios. Some of these combinations can cause spanning tree to put the ports on the channeling side into **errdisable** state (that is, shut them down).

Table 22-6: Channeling Mode Scenarios

Switch-A Channel Mode	Switch-B Channel Mode	Channel State
On	On	Channel
On	Off	Not Channel (errdisable)
On	Auto	Not Channel (errdisable)
On	Desirable	Not Channel (errdisable)
Off	On	Not Channel (errdisable)

Off	Off	Not Channel
Off	Auto	Not Channel
Off	Desirable	Not Channel
Auto	On	Not Channel (errdisable)
Auto	Off	Not Channel
Auto	Auto	Not Channel
Auto	Desirable	Channel
Desirable	On	Not Channel (errdisable)
Desirable	Off	Not Channel
Desirable	Auto	Channel
Desirable	Desirable	Channel

You turned off the channel from the previous example with this command on SwitchA and SwitchB.

```
Switch-A (enable) set port channel 2/1-4 auto
Port(s) 2/1-4 channel mode set to auto.
```

The default channel mode for a port that is able to channel is auto. In order to verify this, enter this command:

```
Switch-A (enable) show port channel 2/1
Port  Status      Channel  Channel  Neighbor  Neighbor
      mode        status   device   port
-----
2/1   connected  auto    not channel
```

The previous command also shows that currently the ports do not channel. Another way to verify the channel state is this.

```
Switch-A (enable) show port channel
No ports channelling
Switch-B (enable) show port channel
No ports channelling
```

Is really very simple to make the channel work with PAgP. At this point both switches are set to auto mode which means that they channel if a connected port sends a PAgP request to channel. If you set SwitchA to desirable, SwitchA, it causes SwitchA to send PAgP packets to the other switch and asks it to channel.

```
Switch-A (enable) set port channel 2/1-4 desirable
```

```

Port(s) 2/1-4 channel mode set to desirable.
1999 Dec 15 22:03:18 %PAGP-5-PORTFROMSTP:Port 2/1 left bridgl
1999 Dec 15 22:03:18 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
1999 Dec 15 22:03:18 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
1999 Dec 15 22:03:18 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
1999 Dec 15 22:03:19 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
1999 Dec 15 22:03:19 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
1999 Dec 15 22:03:20 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
1999 Dec 15 22:03:23 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1-4
1999 Dec 15 22:03:23 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/1-4
1999 Dec 15 22:03:23 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/1-4
1999 Dec 15 22:03:24 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/1-4

```

In order to view the channel, do this.

```
Switch-A (enable) show port channel
```

Port	Status	Channel mode	Channel status	Neighbor device	Neighbor port
2/1	connected	desirable channel		WS-C5505	066509957(Sw 2/1
2/2	connected	desirable channel		WS-C5505	066509957(Sw 2/2
2/3	connected	desirable channel		WS-C5505	066509957(Sw 2/3
2/4	connected	desirable channel		WS-C5505	066509957(Sw 2/4

Since SwitchB was in auto mode, it responded to the PAgP packets and created a channel with SwitchA.

```
Switch-B (enable)
```

```

2000 Jan 14 20:26:41 %PAGP-5-PORTFROMSTP:Port 2/1 left bridgl
2000 Jan 14 20:26:41 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
2000 Jan 14 20:26:41 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
2000 Jan 14 20:26:41 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
2000 Jan 14 20:26:45 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
2000 Jan 14 20:26:45 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
2000 Jan 14 20:26:45 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
2000 Jan 14 20:26:47 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1-4
2000 Jan 14 20:26:47 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/1-4
2000 Jan 14 20:26:47 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/1-4
2000 Jan 14 20:26:48 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/1-4

```

```
Switch-B (enable) show port channel
```

Port	Status	Channel mode	Channel status	Neighbor device	Neighbor port
2/1	connected	auto channel		WS-C5505	066507453(Sw 2/1
2/2	connected	auto channel		WS-C5505	066507453(Sw 2/2
2/3	connected	auto channel		WS-C5505	066507453(Sw 2/3
2/4	connected	auto channel		WS-C5505	066507453(Sw 2/4

Note: It is recommended to set both sides of the channel to **desirable** so that both sides try to initiate the channel if one side drops out. If you set the EtherChannel ports on SwitchB to **desirable** mode, even though the channel is currently active and in **auto** mode, it poses no problem. This is the command.

```
Switch-B (enable) set port channel 2/1-4 desirable
Port(s) 2/1-4 channel mode set to desirable.
```

```
Switch-B (enable) show port channel
```

Port	Status	Channel mode	Channel status	Neighbor device	Neighbor port
2/1	connected	desirable	channel	WS-C5505 066507453(Sw	2/1
2/2	connected	desirable	channel	WS-C5505 066507453(Sw	2/2
2/3	connected	desirable	channel	WS-C5505 066507453(Sw	2/3
2/4	connected	desirable	channel	WS-C5505 066507453(Sw	2/4

Now, if SwitchA drops out for some reason, or if new hardware replaces SwitchA, SwitchB tries to re-establish the channel. If the new equipment cannot channel, SwitchB treats its ports 2/1-4 as normal non-channelling ports. This is one of the benefits of the usage of the **desirable** mode. If the channel was configured with the PAgP on mode and one side of the connection has an error of some kind or a reset, it can cause an **errdisable** state (shutdown) on the other side. With PAgP set in desirable mode on each side, the channel stabilizes and renegotiates the EtherChannel connection.

Trunking and EtherChannel

EtherChannel is independent of trunking. You can turn trunking on, or you can leave trunking off. You can also turn trunking on for all the ports before you create the channel, or you can turn it on after you create the channel (as you do here). As far as EtherChannel is concerned, it does not matter; trunking and EtherChannel are completely separate features. What does matter is that all the ports involved are in the same mode: either they are all trunking before you configure the channel, or they are all not trunking before you configure the channel. All the ports must be in the same trunking state before you create the channel. Once a channel is formed, whatever is changed on one port is also changed for the other ports in the channel. The modules used in this test bed can do ISL or 802.1q trunking. By default, the modules are set to auto trunking and negotiate mode, which means that they trunk if the other side asks them to trunk, and they negotiate whether to use the ISL or 802.1q method for trunking. If not asked to trunk, they work as normal non-trunking ports.

```
Switch-A (enable) show trunk 2
```

Port	Mode	Encapsulation	Status	Native vlan
2/1	auto	negotiate	not-trunking	1
2/2	auto	negotiate	not-trunking	1
2/3	auto	negotiate	not-trunking	1
2/4	auto	negotiate	not-trunking	1

There are a number of different ways to turn on trunking. For this example, you set SwitchA to desirable. SwitchA is already set to negotiate. The combination desirable/negotiate causes SwitchA to ask SwitchB to trunk and to negotiate the type of trunking to do (ISL or 802.1q). Since SwitchB defaults to autonegotiate, SwitchB responds to the request of SwitchA. These results occur:

```
Switch-A (enable) set trunk 2/1 desirable
Port(s) 2/1-4 trunk mode set to desirable.
Switch-A (enable)
1999 Dec 18 20:46:25 %DTP-5-TRUNKPORTON:Port 2/1 has become isl trunk
1999 Dec 18 20:46:25 %DTP-5-TRUNKPORTON:Port 2/2 has become isl trunk
1999 Dec 18 20:46:25 %PAGP-5-PORTFROMSTP:Port 2/1 left bridge port 2/1-4
1999 Dec 18 20:46:25 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/1-4
1999 Dec 18 20:46:25 %DTP-5-TRUNKPORTON:Port 2/3 has become isl trunk
1999 Dec 18 20:46:26 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/1-4
1999 Dec 18 20:46:26 %DTP-5-TRUNKPORTON:Port 2/4 has become isl trunk
1999 Dec 18 20:46:26 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/1-4
1999 Dec 18 20:46:28 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1-4
1999 Dec 18 20:46:29 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/1-4
1999 Dec 18 20:46:29 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/1-4
1999 Dec 18 20:46:29 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/1-4
```

```
Switch-A (enable) show trunk 2
```

Port	Mode	Encapsulation	Status	Native vlan
2/1	desirable	n-isl	trunking	1
2/2	desirable	n-isl	trunking	1
2/3	desirable	n-isl	trunking	1
2/4	desirable	n-isl	trunking	1

The trunk mode was set to desirable. The result was that the trunking mode was negotiated with the neighbor switch, and they decided on ISL (**n-isl**). The current status now is **trunking**. This is what happened on SwitchB because of the command issued on SwitchA.

```
Switch-B (enable)
2000 Jan 17 19:09:52 %DTP-5-TRUNKPORTON:Port 2/1 has become isl trunk
2000 Jan 17 19:09:52 %DTP-5-TRUNKPORTON:Port 2/2 has become isl trunk
2000 Jan 17 19:09:52 %PAGP-5-PORTFROMSTP:Port 2/1 left bridge port 2/1-4
2000 Jan 17 19:09:52 %DTP-5-TRUNKPORTON:Port 2/3 has become isl trunk
2000 Jan 17 19:09:52 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/1-4
2000 Jan 17 19:09:53 %DTP-5-TRUNKPORTON:Port 2/4 has become isl trunk
2000 Jan 17 19:09:53 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/1-4
2000 Jan 17 19:09:53 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/1-4
2000 Jan 17 19:09:55 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1-4
2000 Jan 17 19:09:55 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/1-4
2000 Jan 17 19:09:55 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/1-4
2000 Jan 17 19:09:55 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/1-4
```

```
Switch-B (enable) show trunk 2
```

Port	Mode	Encapsulation	Status	Native vlan
2/1	auto	n-isl	trunking	1
2/2	auto	n-isl	trunking	1
2/3	auto	n-isl	trunking	1
2/4	auto	n-isl	trunking	1

Notice that all four ports (2/1-4) became trunks, even though you only specifically changed one port (2/1) to desirable. This is an example of how the change of one port in the channel affects all the ports.

Troubleshoot EtherChannel

The challenges for EtherChannel can be divided into two main areas: troubleshoot the issue within the configuration phase and troubleshoot the issue within the execution phase. Configuration errors usually occur because of mismatched parameters on the ports involved (different speeds, different duplex, different spanning tree port values, and so on). You can also generate errors within the configuration if you set the channel on one side to **on** and wait too long before you configure the channel on the other side. This causes spanning tree loops, which generates an error, and shuts down the port.

When an error is encountered while you configure EtherChannel, be sure to check the status of the ports after you correct the EtherChannel error situation. If the port status is *errdisable*, that means the ports have been shut down by the software and they do not come on again until you enter the **set port enable** command.

Note: If the port status becomes *errdisable*, you must specifically enable the ports with the **set port enable** command for the ports to become active. Currently, you can correct all the EtherChannel issues, but the ports do not come up or form a channel until they are enabled again! Future versions of the operating system can periodically check if *errdisable* ports must be enabled.

For these tests you turn trunking and EtherChannel off: Mismatched Parameters; Wait Too Long Before You Configure the Other Side; Correct Errdisable State; and Show What Happens When a Link Breaks and is Restored.

Mismatched Parameters

Here is an example of mismatched parameters. you set port 2/4 in VLAN 2 while the other ports are still in VLAN 1. In order to create a new VLAN, you must assign a VTP domain for the switch and create the VLAN.

```
Switch-A (enable) show port channel
No ports channelling
```

```
Switch-A (enable) show port
```

Port	Name	Status	Vlan	Level	Duplex	Speed	Type
2/1		connected	1	normal	a-full	a-100	10/100BaseTX
2/2		connected	1	normal	a-full	a-100	10/100BaseTX
2/3		connected	1	normal	a-full	a-100	10/100BaseTX
2/4		connected	1	normal	a-full	a-100	10/100BaseTX

Switch-A (enable) **set vlan 2**
Cannot add/modify VLANs on a VTP server without a domain name.

Switch-A (enable) **set vtp domain testDomain**
VTP domain testDomain modified

Switch-A (enable) **set vlan 2 name vlan2**
Vlan 2 configuration successful

Switch-A (enable) **set vlan 2 2/4**
VLAN 2 modified.
VLAN 1 modified.
VLAN Mod/Ports

```
-----  
2      2/4
```

Switch-A (enable)
1999 Dec 19 00:19:34 %PAGP-5-PORTFROMSTP:Port 2/4 left bridg4

Switch-A (enable) **show port**

Port	Name	Status	Vlan	Level	Duplex	Speed	Type
2/1		connected	1	normal	a-full	a-100	10/100BaseTX
2/2		connected	1	normal	a-full	a-100	10/100BaseTX
2/3		connected	1	normal	a-full	a-100	10/100BaseTX
2/4		connected	2	normal	a-full	a-100	10/100BaseTX

Switch-A (enable) **set port channel 2/1-4 desirable**
Port(s) 2/1-4 channel mode set to desirable.

Switch-A (enable)
1999 Dec 19 00:20:19 %PAGP-5-PORTFROMSTP:Port 2/1 left bridge port 2/1
1999 Dec 19 00:20:19 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
1999 Dec 19 00:20:19 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
1999 Dec 19 00:20:20 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
1999 Dec 19 00:20:20 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
1999 Dec 19 00:20:22 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
1999 Dec 19 00:20:22 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
1999 Dec 19 00:20:24 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1-2
1999 Dec 19 00:20:25 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/1-2
1999 Dec 19 00:20:25 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/3
1999 Dec 19 00:20:25 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/4

Switch-A (enable) **show port channel**

Port	Status	Channel mode	Channel status	Neighbor device	Neighbor port
2/1	connected	desirable	channel	WS-C5505	066509957(Sw 2/1
2/2	connected	desirable	channel	WS-C5505	066509957(Sw 2/2

Notice that the channel only formed between ports 2/1-2. Ports 2/3-4 were left out because port 2/4 was in a different VLAN. There was no error message; PAgP just did what it could to make the

channel work. You need to watch the results when you create the channel to make sure it did what you wanted it to do.

Now set the channel manually to "on" with port 2/4 in a different VLAN and see what happens. First you set the channel mode back to auto in order to tear down the current channel, then you set the channel manually to "on".

```
Switch-A (enable) set port channel 2/1-4 auto
Port(s) 2/1-4 channel mode set to auto.
Switch-A (enable)
1999 Dec 19 00:26:08 %PAGP-5-PORTFROMSTP:Port 2/1 left bridge port 2/1-2
1999 Dec 19 00:26:08 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/1-2
1999 Dec 19 00:26:08 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
1999 Dec 19 00:26:08 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
1999 Dec 19 00:26:18 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1
1999 Dec 19 00:26:19 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/2
1999 Dec 19 00:26:19 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/3
1999 Dec 19 00:26:19 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/4
```

```
Switch-A (enable) show port channel
No ports channelling
```

```
Switch-A (enable) set port channel 2/1-4 on
Mismatch in vlan number.
Failed to set port(s) 2/1-4 channel mode to on.
```

```
Switch-A (enable) show port channel
No ports channelling
```

On SwitchB you can turn the channel on and notice that it says the ports channel fine, but you know that SwitchA is not configured correctly.

```
Switch-B (enable) show port channel
No ports channelling
```

```
Switch-B (enable) show port
Port Name Status Vlan Level Duplex Speed Type
-----
2/1 connected 1 normal a-full a-100 10/100BaseTX
2/2 connected 1 normal a-full a-100 10/100BaseTX
2/3 connected 1 normal a-full a-100 10/100BaseTX
2/4 connected 1 normal a-full a-100 10/100BaseTX
```

```
Switch-B (enable) set port channel 2/1-4 on
Port(s) 2/1-4 channel mode set to on.
```

```
Switch-B (enable)
2000 Jan 17 22:54:59 %PAGP-5-PORTFROMSTP:Port 2/1 left bridge port 2/1
2000 Jan 17 22:54:59 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/2
2000 Jan 17 22:54:59 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
2000 Jan 17 22:54:59 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
```

```

2000 Jan 17 22:55:00 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1-4
2000 Jan 17 22:55:00 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/1-4
2000 Jan 17 22:55:00 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/1-4
2000 Jan 17 22:55:00 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/1-4

```

```
Switch-B (enable) show port channel
```

Port	Status	Channel mode	Channel status	Neighbor device	Neighbor port
2/1	connected	on	channel	WS-C5505 066507453(Sw	2/1
2/2	connected	on	channel	WS-C5505 066507453(Sw	2/2
2/3	connected	on	channel	WS-C5505 066507453(Sw	2/3
2/4	connected	on	channel	WS-C5505 066507453(Sw	2/4

This makes it clear that you must check both sides of the channel when you manually configure the channel to make sure that both sides are up, not just one side. This output shows that SwitchB is set for a channel, but SwitchA does not channel because it has one port that is in the wrong VLAN.

Wait Too Long Before You Configure the Other Side

In this situation, SwitchB has EtherChannel turned on, but SwitchA does not because it has a VLAN configuration error (ports 2/1-3 are in vlan1, port 2/4 is in vlan2). Here is what happens when one side of a EtherChannel is set to on while the other side is still in auto mode. SwitchB, after a few minutes, shut down its ports because of a spanning loop detection. This is because SwitchB ports 2/1-4 all act like one big port while SwitchA ports 2/1-4 are all totally independent ports. A broadcast sent from SwitchB to SwitchA on port 2/1 is sent back to SwitchB on ports 2/2, 2/3 and 2/4 because SwitchA treats these ports as independent ports. This is why SwitchB reports that there is a spanning tree loop. Notice that the ports on SwitchB are now disabled and have a status of *errdisable*.

```
Switch-B (enable)
```

```

2000 Jan 17 22:55:48 %SPANNTREE-2-CHNMISCFG: STP loop - channel 2/1-4 is disabled in
vlan 1.
2000 Jan 17 22:55:49 %PAGP-5-PORTFROMSTP:Port 2/1 left bridge port 2/1-4
2000 Jan 17 22:56:01 %PAGP-5-PORTFROMSTP:Port 2/2 left bridge port 2/1-4
2000 Jan 17 22:56:13 %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/1-4
2000 Jan 17 22:56:36 %PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/1-4

```

```
Switch-B (enable) show port channel
```

Port	Status	Channel mode	Channel status	Neighbor device	Neighbor port
2/1	errdisable	on	channel		
2/2	errdisable	on	channel		
2/3	errdisable	on	channel		
2/4	errdisable	on	channel		

```
Switch-B (enable) show port
```

Port	Name	Status	Vlan	Level	Duplex	Speed	Type
------	------	--------	------	-------	--------	-------	------

```

2/1          errdisable 1          normal  auto  auto 10/100BaseTX
2/2          errdisable 1          normal  auto  auto 10/100BaseTX
2/3          errdisable 1          normal  auto  auto 10/100BaseTX
2/4          errdisable 1          normal  auto  auto 10/100BaseTX

```

Correct Errdisable State

Sometimes when you try to configure EtherChannel, but the ports are not configured the same, it causes the ports on one side of the channel or the other to be shut down. The link lights are yellow on the port. You can tell this by the console if you type **show port**. The ports are listed as **errdisable**. In order to recover from this, you must fix the mis-matched parameters on the ports involved, then re-enable the ports. Just note that to re-enable the ports is a separate step that must be done for the ports to become functional again.

In this example you know SwitchA had a vlan mismatch. you go to SwitchA and put port 2/4 back in to vlan1. Then you turn the channel for ports 2/1-4 on. SwitchA does not show connected until you re-enable SwitchB ports. Then when you have fixed SwitchA and put it in channelling mode, you go back to SwitchB and re-enable the ports.

```
Switch-A (enable) set vlan 1 2/4
```

```
VLAN 1 modified.
```

```
VLAN 2 modified.
```

```
VLAN Mod/Ports
```

```
-----
1      2/1-24
```

```
Switch-A (enable) set port channel 2/1-4 on
```

```
Port(s) 2/1-4 channel mode set to on.
```

```
Switch-A (enable) sh port channel
```

```

Port  Status      Channel  Channel  Neighbor  Neighbor
      mode         status   device   port
-----
2/1   notconnect on       channel
2/2   notconnect on       channel
2/3   notconnect on       channel
2/4   notconnect on       channel
-----

```

```
Switch-B (enable) show port channel
```

```

Port  Status      Channel  Channel  Neighbor  Neighbor
      mode         status   device   port
-----
2/1   errdisable on       channel
2/2   errdisable on       channel
2/3   errdisable on       channel
2/4   errdisable on       channel
-----

```

```
Switch-B (enable) set port enable 2/1-4
```

```
Ports 2/1-4 enabled.
```

```
Switch-B (enable) 2000 Jan 17 23:15:22 %PAGP-5-PORTTOSTP:Port 2/1 joined bridg4
```

```
2000 Jan 17 23:15:22 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/1-4
```

```
2000 Jan 17 23:15:22 %PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/1-4
2000 Jan 17 23:15:22 %PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/1-4
```

```
Switch-B (enable) show port channel
```

Port	Status	Channel mode	Channel status	Neighbor device	Neighbor port
2/1	connected	on	channel		
2/2	connected	on	channel		
2/3	connected	on	channel		
2/4	connected	on	channel		

Show What Happens When a Link Breaks and is Restored

When a port in the channel goes down, any packets that are normally sent on that port are shifted over to the next port in the channel. You can verify this happens with the **show mac** command. In this test bed, you have SwitchA send ping packets to SwitchB in order to see which link the traffic uses. First you clear the counters, then show mac, send three pings, and then **show mac** again to see on which channel the ping responses were received.

```
Switch-A (enable) clear counters
```

```
This command will reset all MAC and port counters reported in CLI and SNMP.
```

```
Do you want to continue (y/n) [n]? y
```

```
MAC and Port counters cleared.
```

```
Switch-A (enable) show port channel
```

Port	Status	Channel mode	Channel status	Neighbor device	Neighbor port
2/1	connected	on	channel	WS-C5505	066509957(Sw 2/1
2/2	connected	on	channel	WS-C5505	066509957(Sw 2/2
2/3	connected	on	channel	WS-C5505	066509957(Sw 2/3
2/4	connected	on	channel	WS-C5505	066509957(Sw 2/4

```
Switch-A (enable) show mac
```

Port	Rcv-Unicast	Rcv-Multicast	Rcv-Broadcast
2/1		0	18
2/2		0	2
2/3		0	2
2/4		0	2

```
Switch-A (enable) ping 172.16.84.17
```

```
172.16.84.17 is alive
```

```
Switch-A (enable) ping 172.16.84.17
```

```
172.16.84.17 is alive
```

```
Switch-A (enable) ping 172.16.84.17
```

```
172.16.84.17 is alive
```

```
Switch-A (enable) show mac
```

Port	Rcv-Unicast	Rcv-Multicast	Rcv-Broadcast
2/1	3	24	0
2/2	0	2	0
2/3	0	2	0
2/4	0	2	0

At this point, you have received the ping responses on port 3/1. When SwitchB console sends a response to SwitchA, the EtherChannel uses port 2/1. Now you shut down port 2/1 on SwitchB. From SwitchA you issue another ping and see what channel the response comes back on. (SwitchA sends on the same port to which SwitchB is connected. you just show the received packets from SwitchB because the transmit packets are further down in the **show mac** display).

```
1999 Dec 19 01:30:23 %PAGP-5-PORTFROMSTP:Port 2/1 left bridge port 2/1-4
```

```
Switch-A (enable) ping 172.16.84.17
```

```
172.16.84.17 is alive
```

```
Switch-A (enable) show mac
```

Port	Rcv-Unicast	Rcv-Multicast	Rcv-Broadcast
2/1	3	37	0
2/2	1	27	0
2/3	0	7	0
2/4	0	7	0

Now that port 2/1 is disabled, EtherChannel automatically uses the next port in the channel, 2/2. Now you re-enable port 2/1 and wait for it to join the bridge group. you then issue two more pings.

```
1999 Dec 19 01:31:33 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1-4
```

```
Switch-A (enable) ping 172.16.84.17
```

```
172.16.84.17 is alive
```

```
Switch-A (enable) ping 172.16.84.17
```

```
172.16.84.17 is alive
```

```
Switch-A (enable) show mac
```

Port	Rcv-Unicast	Rcv-Multicast	Rcv-Broadcast
2/1	5	50	0
2/2	1	49	0
2/3	0	12	0
2/4	0	12	0

Note that these pings are sent from port 2/1. When the link comes back up, EtherChannel again adds it to the bundle and uses it. All this is done transparently to the user.

Commands Used in this Section

These are the commands that were used in this section.

Commands to Use to Set the Configuration

set port channel on- to turn on the EtherChannel feature.

set port channel auto- to reset the ports to their default mode of auto.

set port channel desirable- to send PAgP packets to the other side requests for a channel to be created.

set port enable- to enable the ports after set port disable or after an errdisable state.

set port disable- to disable a port while other configuration settings are made.

set trunk desirable- to turn on trunking and cause this port to send a request to the other switch to indicate that this is a trunk link. If the port is set to negotiate (the default setting) to negotiate the type of trunking to use on the link (ISL or 802.1q).

Commands to Use to Verify the Configuration

show version- to display what version of software the switch runs.

show module- to display which modules are installed in the switch.

show port capabilities- to determine if the ports you want to use have the ability to do EtherChannel.

show port- to determine the status of the port (notconnect, connected) and speed and duplex settings.

ping- to test connectivity to the other switch.

show port channel- to see the current status of the EtherChannel bundle.

show port channel mod/port- to give a more detailed view of the channel status of a single port.

show spantree- to verify that spanning tree looked at the channel as one link.

show trunk- to see the trunking status of ports.

Commands to Use to Troubleshoot the Configuration

show port channel- to see the current status of the EtherChannel bundle.

show port- to determine the status of the port (notconnect, connected) and speed and duplex settings.

clear counters- to reset the switch packet counters to zero. The counters are visible with the **show mac** command.

show mac- to view packets received and sent by the switch.

ping- to test connectivity to the other switch and generate traffic that shows up with the **show mac** command.

Use Portfast and Other Commands to Fix End-Station Startup Connectivity Problems

If you have workstations connected to switches which are unable to login to your network domain (NT or Novell), or are unable to get a DHCP address, then you can try the suggestions listed in this document before you explore other avenues. The suggestions are relatively easy to implement and are very often the cause of workstation connectivity problems encountered through the workstation initialization/startup phase.

With more and more users that deploy switching to the desktop and replace their shared hubs with switches, you often see problems introduced in client/server environments because of this initial delay. The biggest problem you see is that Windows 95/98/NT, Novell, VINES, IBM NetworkStation/IBM Thin Clients, and AppleTalk clients are unable to connect to their servers. If the software on these devices is not persistent within the startup procedure they no longer try to connect to their server before the switch has even allowed traffic to pass through.

Note: This initial connectivity delay often manifests itself as errors that appear when you first boot up a workstation. These are several examples of error messages and errors you can see:

A Microsoft networking client displays, "No Domain Controllers Available."

DHCP reports, "No DHCP Servers Available."

A Novell IPX networking workstation does not have the "Novell Login Screen" upon bootup.

An AppleTalk networking client displays, "Access to your AppleTalk network has been interrupted. In order to re-establish your connection, open and close the AppleTalk control panel." It is also possible that the AppleTalk Client Chooser application either do not display a

zone list or display an incomplete zone list.

The initial connectivity delay is also frequently seen in a switched environment in which a network administrator updates software or drivers. In this case, a vendor can optimize the drivers so that network initialization procedures happen earlier in the startup process of the client (before the switch is ready to process the packets).

With the various features that are now included in some switches, it can take close to a minute for a switch to begin to service a newly connected workstation. This delay can affect the workstation every time it is turned on or rebooted. These are the four main features that cause this delay:

Spanning-Tree Protocol (STP)

EtherChannel negotiation

Trunking negotiation

Link speed/duplex negotiation between the switch and the workstation

The four features are listed in order of which cause the most delay (Spanning-Tree Protocol) to which cause the least delay (speed/duplex negotiation). A workstation connected to a switch usually does not cause spanning tree loops, usually does not need EtherChannel, and usually does not need to negotiate a trunking method. (If you disable link speed/detection negotiation, it can also reduce port delay if you need to optimize your startup time as much as possible.)

This section shows how to implement startup speed-optimization commands on three Catalyst switch platforms. In the timing sections, you show how the switch port delay is reduced, and by how much.

Contents

[Background](#)

[How to Reduce Startup Delay on the Catalyst 4000/5000/6000 Switch](#)

[Timing Tests on the Catalyst 5000](#)

[How to Reduce Startup Delay on the Catalyst 2900XL/3500XL Switch](#)

[Timing Tests on the Catalyst 2900XL](#)

[How to Reduce Startup Delay on the Catalyst 1900/2800 Switch](#)

[Timing Test on the Catalyst 2820](#)

[An Additional Benefit to Portfast](#)

The terms "workstation", "end-station", "server" are all used interchangeably in this section. What you refer to is any device directly connected to a switch by a single NIC card. It can also refer to devices with multiple NIC cards where the NIC card is only used for redundancy, in other words the workstation or server is not configured to act as a bridge, it just has multiple NIC cards for redundancy.

Note: There are some server NIC cards that support trunking and/or EtherChannel. There are situations where the server needs to live on several VLANs at the same time (trunking) or the server needs more bandwidth on the link that connects it to the switch (EtherChannel). In such cases, do not turn PAgP off and do not turn trunking off. Also, these devices are rarely turned off or reset. The instructions in this document do not apply to these type of devices.

Background

This section covers four features which some switches have that cause initial delays when a device is connected to a switch. Usually, a workstation either does not cause the spanning tree problem (loops), or does not need the feature (PAgP, DTP), so the delay is unnecessary.

Spanning Tree

If you have recently started to move from a hub environment to a switch environment these connectivity problems can show up because a switch works much differently than a hub. A switch provides connectivity at the datalink layer, not at the physical layer. The switch has to use a bridging algorithm to decide if packets received on a port need to be transmitted out other ports. The bridging algorithm is susceptible to physical loops in the network topology. Because of this susceptibility to loops, switches run a protocol called the spanning tree protocol (STP) that causes loops to be eliminated in the topology. When STP runs it causes all ports that are included in the spanning tree process to become active much slower than they otherwise would, as it detects and blocks loops. A bridged network that has physical loops, without spanning tree, breaks. In spite of the time involved, STP is a good thing. The spanning tree that runs on Catalyst switches is an industry standard specification (IEEE 802.1d).

After a port on the switch has link and joins the bridge group it runs spanning tree on that port. A port that runs spanning tree can have 1 of 5 states: Blocking, Listening, Learning, Forwarding, and Disabled. Spanning tree dictates that the port starts out blocking, then immediately moves through the listening and learning phases. By default, it spends approximately 15 seconds listening and 15 seconds learning.

While in the listening state, the switch tries to determine where it fits in the spanning tree topology. It especially wants to know whether this port is part of a physical loop. If it is part of a loop, this port can be chosen to go into blocking mode. Blocking means it does not send or receive user data in order to eliminate loops. If the port is not part of a loop, it proceeds to the learning state which involves learning which MAC addresses live off of this port. This whole spanning tree initialization process takes about 30 seconds.

If you connect a workstation or a server with a single NIC card to a switch port, this connection cannot create a physical loop. These connections are considered leaf nodes. There is no reason to make the workstation wait 30 seconds while the switch checks for loops when the workstation cannot cause a loop. So, Cisco added a feature called "Portfast" or "Fast-Start", which means the spanning tree for this port can assume that the port is not part of a loop and can immediately move to the forwarding state, and skip the blocking, listening, or learning states. This can save a lot of

time. This command does not turn spanning tree off. It just makes spanning tree on the selected port skip a few (unnecessary in this circumstance) steps in the beginning.

Note: The Portfast feature must never be used on switch ports that connect to other switches or hubs or routers. These connections can cause physical loops, and it is very important that spanning tree go through the full initialization procedure in these situations. A spanning tree loop can bring your network down. If portfast is turned on for a port that is part of a physical loop, it can cause a window of time where packets could possibly be continuously forwarded (and even multiply) in such a way that the network cannot recover. In later Catalyst operating system software (5.4(1)), there is a feature called Portfast BPDU-Guard, which detects the reception of BPDUs on ports that have Portfast enabled. Since this must never happen, BPDU-Guard puts the port into "errDisable" state.

EtherChannel

Another feature a switch can have is called EtherChannel (or Fast EtherChannel, or Gigabit EtherChannel). This feature allows multiple links between the same two devices to work as if they were one fast link, with traffic load balanced among the links. A switch can form these bundles automatically with a neighbor with a protocol called Port Aggregation Protocol (PAgP). Switch ports that can run PAgP usually default to a passive mode called "auto" which means that they can form a bundle if the neighbor device across the link asks them to. If you run the protocol in auto mode, it can cause a port to delay for up to 15 seconds before it passes control to the spanning tree algorithm (PAgP runs on a port before spanning tree does). There is no reason for PAgP to run on a port connected to a workstation. If you set the switch port PAgP mode to "off," it eliminates this delay.

Trunking

Another switch feature is the ability of a port to form a trunk. A trunk is configured between two devices when they need to carry traffic from multiple Virtual Local Area Networks (VLANs). A VLAN is something switches create to make a group of workstations appear to be on their own "segment" or "broadcast domain." Trunk ports make these VLANs extend across multiple switches, so that a single VLAN can cover an entire campus. They do this with the addition of tags to the packets; this indicates to which VLAN the packet belongs.

There are different types of trunking protocols. If a port can become a trunk, it can also have the ability to trunk automatically, and in some cases even negotiate what type of trunking to use on the port. This ability to negotiate the trunking method with the other device is called Dynamic Trunking Protocol (DTP), the precursor to DTP is a protocol called Dynamic ISL (DISL). If these protocols run they can delay a port on the switch that becomes active.

Usually, a port connected to a workstation belongs to only one VLAN, and therefore does not need to trunk. If a port has the ability to negotiate the formation of a trunk it usually defaults to the "auto" mode. If the port is changed to a trunking mode of "off" it further reduces the delay of a switch port that becomes active.

Speed and Duplex Negotiation

All you need to do is turn on Portfast and turn off PAgP (if present) to solve the problem, but if you need to eliminate every possible second you could also set the port speed and duplex manually on the switch if it is a multi-speed port (10/100). Auto-negotiation is a nice feature but if you turn it off you could save 2 seconds on a Catalyst 5000 (It does not help much on the 2800 or 2900XL).

There can be complications, though, if you turn off auto-negotiation on the switch but leave it active on the workstation. Since the switch does not negotiate with the client, the client can choose

the same duplex setting that the switch uses or not. See the "Troubleshoot Ethernet 10/100Mb Half/Half/Full Duplex Auto-Negotiation" for additional information on the caveats of auto-negotiation.

How to Reduce Startup Delay on the Catalyst 4000/5000/6000 Switch

These five commands show how to turn on Portfast, how to turn off PAgP negotiation, turn off trunking negotiation (DISL, DTP), and turn off speed/duplex negotiation. The **set spantree portfast** command is done on a range of ports at once (**set spantree portfast 2/1-12 enable**). Usually **set port channel** must be turned off with a valid group of channel-capable ports. In this case module two has the ability to channel with ports 2/1-2 or with ports 2/1-4, so either of these groups of ports would have been valid to use.

Note: Version 5.2 of Cat OS for Catalyst 4000/5000 has a new command called **set port host** which is a macro that combines these commands into one easy-to-use command (except it does not change the speed and duplex settings).

Configuration

```
Switch-A (enable) set spantree portfast 2/1 enable
```

```
Warning: Spantree port fast start should only be enabled on ports connected  
to a single host. Connecting hubs, concentrators, switches, bridges, and so on to  
a fast start port can cause temporary spanning tree loops. Use with caution.
```

```
Spantree port 2/1 fast start enabled.
```

```
Switch-A (enable) set port channel 2/1-2 off
```

```
Port(s) 2/1-2 channel mode set to off.
```

```
Switch-A (enable) set trunk 2/1 off
```

```
Port(s) 2/1 trunk mode set to off.
```

The changes to the configuration are automatically saved to NVRAM.

Verification

The version of the switch software used in this document is 4.5(1). For the full output of show version and show module refer to this timing test section.

```
Switch-A (enable) show version  
WS-C5505 Software,  
Version McpSW: 4.5(1) NmpSW: 4.5(1)
```

This command shows how to view the current state of a port with regards to spanning tree. Currently the port is in the spanning tree forwarding state (sending and receiving packets) and the Fast-Start column shows that portfast is currently disabled. In other words, the port can take at least 30 seconds to move to the forwarding state whenever it initializes.

```
Switch-A (enable) show port spantree 2/1
```

Port	Vlan	Port-State	Cost	Priority	Fast-Start	Group-Method
2/1	1	forwarding	19	32		

disabled

Now you enable portfast on this switch port. The switch warns us that this command must only be used on ports that are connected to a single host (a workstation, server, and so on) and never to be used on ports connected to other hubs or switches. The reason you enable portfast is, so the port start to forward immediately. You can do this because a workstation or server does not cause a network loop. This can waste time. But another hub or switch can cause a loop, and you want to always go through the normal listening and learning stages when you connect to these types of devices.

```
Switch-A (enable) set spantree portfast 2/1 enable
```

Warning: Spantree port fast start should only be enabled on ports connected to a single host. Connecting hubs, concentrators, switches, bridges, and so on to a fast start port can cause temporary spanning tree loops. Use with caution.

Spantree port 2/1 fast start enabled.

In order to verify that Portfast is enabled for this port do this command.

```
Switch-A (enable) show port spantree 2/1
```

Port	Vlan	Port-State	Cost	Priority	Fast-Start	Group-Method
2/1	1	forwarding	19	32		

enabled

Another way to view the Portfast settings for one or more ports is to view the spanning tree information for a specific VLAN. Later on, in the timing section of this document, you show how to have the switch report each stage of spanning tree that it moves through in real time. This output also shows the forward delay time (15 seconds). This is how long spanning tree can be in the listening state and how long it can be in the learning state for each port in the VLAN.

```
Switch-A (enable) show spantree 1
```

VLAN 1

Spanning tree enabled

Spanning tree type ieee

Designated Root 00-e0-4f-94-b5-00

Designated Root Priority 8189

Designated Root Cost 19

Designated Root Port 2/24

Root Max Age 20 sec Hello Time 2 sec **Forward Delay 15 sec**

```

Bridge ID MAC ADDR          00-90-92-b0-84-00
Bridge ID Priority          32768
Bridge Max Age 20 sec      Hello Time 2 sec      Forward Delay 15 sec

```

```

Port      Vlan  Port-State      Cost  Priority  Fast-Start  Group-Method
-----
2/1      1    forwarding      19    32    enabled
...

```

In order to verify that PAgP is off, use the **show port channel** command. Be sure and specify the module number (2 in this case) so that the command shows you the channel mode even if there is no channel formed. If you do **show port channel** with no channels formed, it just says no ports channeling. you want to go further and see the current channel mode.

```

Switch-A (enable) show port channel
No ports channeling

```

```

Switch-A (enable) show port channel 2
Port  Status      Channel  Channel  Neighbor  Neighbor
      mode      status  device   port
-----
2/1  notconnect  auto    not channel
2/2  notconnect  auto    not channel
...

```

```

Switch-A (enable) set port channel 2/1-2 off
Port(s) 2/1-2 channel mode set to off.

```

```

Switch-A (enable) show port channel 2
Port  Status      Channel  Channel  Neighbor  Neighbor
      mode      status  device   port
-----
2/1  connected  off     not channel
2/2  connected  off     not channel
...

```

In order to verify that Trunking negotiation is off, use the **set trunk off** command. You show the default state . Then you turn trunking to off and show the result. You specify module number 2 so that you can see the current channel mode for the ports in this module.

```

Switch-A (enable) show trunk 2
Port  Mode      Encapsulation  Status      Native vlan
-----
2/1   auto     negotiate      not-trunking  1
2/2   auto     negotiate      not-trunking  1
...

```

```

Switch-A (enable) set trunk 2/1-2 off
Port(s) 2/1-2 trunk mode set to off.

```

```
Switch-A (enable) show trunk 2
```

Port	Mode	Encapsulation	Status	Native vlan
2/1	off	negotiate	not-trunking	1
2/2	off	negotiate	not-trunking	1

It is not necessary except in the rarest of cases to turn off speed/duplex auto-negotiation or manually set the speed and duplex on the switch. you give an example of how to do this in the Timing Tests With and Without DTP, PAgP, and Portfast on a Catalyst 5000 section if you feel it is necessary for your situation.

Timing Tests With and Without DTP, PAgP, and Portfast on a Catalyst 5000

This test shows what happens with switch port initialization timing as the various commands are applied. The default settings of the port are used first to give a benchmark. They have portfast disabled, PAgP (EtherChannel) mode is set to auto (it channels if asked to channel), and the trunking mode (DTP) is set to auto (it trunks if asked to trunk). The test then proceeds to turn portfast on and measure the time, then turn PAgP to off and measure the time, then turn trunking off and measure the time. Finally, you turn auto-negotiation off and measure the time. All of these tests are done on a Catalyst 5000 with a 10/100 Fast Ethernet card that supports DTP and PAgP.

Note: When the portfast is turned on it is not the same thing as turning spanning tree off (as noted in the document). With portfast on, spanning tree still runs on the port; it just does not block, listen, or learn, and goes immediately to the forwarding state. Turning spanning tree off is not recommended because it affects the entire VLAN and can leave the network vulnerable to physical topology loops, which can cause serious network problems.

Show the switch Cisco IOS version and configuration (**show version, show module**).

```
Switch-A (enable) show version
```

```
WS-C5505 Software, Version McpSW: 4.5(1) NmpSW: 4.5(1)
```

```
Copyright (c) 1995-1999 by Cisco Systems
```

```
NMP S/W compiled on Mar 29 1999, 16:09:01
```

```
MCP S/W compiled on Mar 29 1999, 16:06:50
```

```
System Bootstrap Version: 3.1.2
```

```
Hardware Version: 1.0 Model: WS-C5505 Serial #: 066507453
```

Mod	Port	Model	Serial #	Versions
1	0	WS-X5530	006841805	Hw : 1.3 Fw : 3.1.2 Fw1: 3.1(2) Sw : 4.5(1)
2	24	WS-X5225R	012785227	Hw : 3.2 Fw : 4.3(1) Sw : 4.5(1)

```

Module Total   Used   Free   Total   Used   Free   Total   Used   Free
-----
1      32640K 13648K 18992K  8192K  4118K  4074K  512K  119K  393K

```

Uptime is 28 days, 18 hours, 54 minutes

Switch-A (enable) **show module**

```

Mod Module-Name      Ports Module-Type      Model      Serial-Num Status
-----
1                      0     Supervisor III    WS-X5530   006841805 ok
2                      24    10/100BaseTX Ethernet WS-X5225R 012785227 ok

```

```

Mod MAC-Address(es)                Hw      Fw      Sw
-----
1  00-90-92-b0-84-00 to 00-90-92-b0-87-ff 1.3     3.1.2   4.5(1)
2  00-50-0f-b2-e2-60 to 00-50-0f-b2-e2-77 3.2     4.3(1)  4.5(1)

```

```

Mod Sub-Type Sub-Model Sub-Serial Sub-Hw
-----
1  NFFC      WS-F5521  0008728786 1.0

```

Set logging for spanning tree to the most verbose (Set logging level spantree 7). This is the default logging level (2) for spanning tree, which means that only critical situations are reported.

Switch-A (enable) show logging

```

Logging buffer size:      500
    timestamp option:    enabled
Logging history size:     1
Logging console:         enabled
Logging server:          disabled
    server facility:     LOCAL7
    server severity:     warnings(4)

```

```

Facility      Default Severity      Current Session Severity
-----
...
spantree      2                      2
...
0(emergencies)  1(alerts)              2(critical)
3(errors)      4(warnings)            5(notifications)
6(information) 7(debugging)

```

The level for spanning tree is changed to 7 (debug), so you can see the spanning tree states change on the port. This configuration change only lasts for the terminal session, then it goes back to normal.

Switch-A (enable) **set logging level spantree 7**

```
System logging facility <spantree for this session set to severity 7(debugging)
```

```
Switch-A (enable) show logging
```

```
...
```

Facility	Default Severity	Current Session Severity
-----	-----	-----
...		
spantree	2	7
...		

Start with the port on the catalyst shut down.

```
Switch-A (enable) set port disable 2/1  
Port 2/1 disabled.
```

Now the time and enable the port. you want to see how long it stays in each state.

```
Switch-A (enable) show time  
Fri Feb 25 2000, 12:20:17  
Switch-A (enable) set port enable 2/1  
Port 2/1 enabled.  
Switch-A (enable)  
2000 Feb 25 12:20:39 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1  
2000 Feb 25 12:20:39 %SPANTREE-6-PORTBLK: port 2/1 state in vlan 1 changed to  
blocking.  
2000 Feb 25 12:20:39 %SPANTREE-6-PORTLISTEN: port 2/1 state in vlane 1 changed  
to Listening.  
2000 Feb 25 12:20:53 %SPANTREE-6-PORTLEARN: port 2/1 state in vlan 1 changed to  
Learning.  
2000 Feb 25 12:21:08 %SPANTREE-6-PORTFWD: port 2/1 state in vlan 1 changed to  
forwarding.
```

Notice from the output that it took about 22 seconds (20:17 to 20:39) for the port to begin the spanning tree blocking stage. This was the time it took to negotiate the link and do DTP and PAgP tasks. When blocking begins, you are now in the spanning tree realm. From blocking the port, it went immediately to listening (20:39 to 20:39). From listening to learning took approximately 14 seconds (20:39 to 20:53).

From learning to forwarding took 15 seconds (20:53 to 21:08). So the total time before the port actually became functional for traffic was about 51 seconds (20:17 to 21:08).

Note: Technically, the listening and learning stage is both be 15 seconds, which is how the forward delay parameter is set for this VLAN. The learning stage probably is closer to 15 seconds than 14 seconds if you had more accurate measurements. None of the measurements here are perfectly accurate. you just tried to give a feel for how long things take.

You know from the output and from the **show spantree** command that spanning tree is active on this port. Let us look at other things that could slow the port as it reaches the forwarding state. The **show port capabilities** command shows that this port has the ability to trunk and to create an EtherChannel. The **show trunk** command says that this port is in auto mode and that it is set to negotiate the type of trunking to use (ISL or 802.1q, negotiated through Dynamic Trunking Protocol (DTP)).

```
Switch-A (enable) show port capabilities 2/1
Model                WS-X5225R
Port                 2/1
Type                 10/100BaseTX

Speed                auto,10,100
Duplex                half,full
Trunk encap type     802.1Q,ISL
Trunk mode          on,off,desirable,auto,nonegotiate
Channel            2/1-2,2/1-4
Broadcast suppression percentage(0-100)
Flow control         receive-(off,on),send-(off,on)
Security             yes
Membership           static,dynamic
Fast start           yes
Rewrite              yes
Switch-A (enable) show trunk 2/1
Port      Mode      Encapsulation  Status      Native vlan
-----  -
2/1      auto      negotiate    not-trunking  1
```

First, you can enable Portfast on the port. Trunking negotiation (DTP) is still in the auto mode, and EtherChannel (PAgP) is still in the auto mode.

```
Switch-A (enable) set port disable 2/1
Port 2/1 disabled.
```

```
Switch-A (enable) set spantree portfast 2/1 enable
```

```
Warning: Spantree port fast start should only be enabled on ports connected
to a single host. Connecting hubs, concentrators, switches, bridges, and so on
to
a fast start port can cause temporary spanning tree loops. Use with caution.
```

```
Spantree port 2/1 fast start enabled.
```

```
Switch-A (enable) show time
```

```
Fri Feb 25 2000, 13:45:23
```

```
Switch-A (enable) set port enable 2/1
```

```
Port 2/1 enabled.
```

```
Switch-A (enable)
```

```
Switch-A (enable)
```

```
2000 Feb 25 13:45:43 %PAGP-5-PORTTOSTP:Port 2/1 joined bridgeport 2/1
```

```
2000 Feb 25 13:45:44 %SPANTREE-6-PORTFWD: port 2/1 state in vlan 1 change to
forwarding.
```

Now you have a total time of **21 seconds!** It takes 20 seconds before it joins the bridge group (45:23 to 45:43). But then, since Portfast is enabled, it only takes one second until STP starts forwarding (instead of 30 seconds). You saved 29 seconds when you enabled Portfast. See if you can reduce the delay further.

Now you turn the PAgP mode to "off." you can see from the show port channel command that the PAgP mode is set to *auto*, which means it channels if asked to by a neighbor that speaks PAgP. You must turn off channeling for at least a group of two ports. You cannot do it for just an individual port.

```
Switch-A (enable) show port channel 2/1
Port  Status      Channel  Channel  Neighbor  Neighbor
      mode         status   device   port
-----
2/1   connected auto    not channel

Switch-A (enable) set port channel 2/1-2 off
Port(s) 2/1-2 channel mode set to off.
```

Shut down the port and repeat the test.

```
Switch-A (enable) set port disable 2/1
Port 2/1 disabled.

Switch-A (enable) show time
Fri Feb 25 2000, 13:56:23
Switch-A (enable) set port enable 2/1
Port 2/1 enabled.
Switch-A (enable)
2000 Feb 25 13:56:32 %PAGP-5-PORTTOSTP:Port 2/1 joined bridgeport 2/1
2000 Feb 25 13:56:32 %SPANTREE-6-PORTFWD: port 2/1 state in vlan 1 changed to
forwarding.
```

Notice that now it only takes **9 seconds** to reach the forwarding state (56:23 to 56:32) instead of 21 seconds as in the previous test. Turning PAgP from *auto* to *off* in this test saved about 12 seconds.

Turn trunking to off (instead of auto) and see how that affects the time it takes for the port to reach forwarding state. You again turn the port off and on and record the time.

```
Switch-A (enable) set trunk 2/1 off
Port(s) 2/1 trunk mode set to off.
Switch-A (enable) set port disable 2/1
```

Port 2/1 disabled.

Start the test with trunking set to off (instead of auto).

```
Switch-A (enable) show time
Fri Feb 25 2000, 14:00:19
Switch-A (enable) set port enable 2/1
Port 2/1 enabled.
Switch-A (enable)
2000 Feb 25 14:00:22 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1
2000 Feb 25 14:00:23 %SPANTREE-6-PORTFWD: port 2/1 state in vlan 1 change for
forwarding.
```

you saved a few seconds at the beginning since it only took **4 seconds** to reach the spanning tree forwarding state (00:19 to 00:22). You saved about 5 seconds when you change the trunking mode from *auto* to *off*.

(Optional) If the switch port initialization time was the problem it must be solved by now. If you have to shave a few more seconds off the time, you could set the port the speed and duplex manually and not use auto-negotiation.

If you set the speed and duplex manually on this side, it requires that you set the speed and duplex on the other side, as well. This is because setting the port speed and duplex disables auto-negotiation on the port, and the device that connects does not see auto-negotiation parameters. The connecting device connects only at half-duplex and the resultant duplex mismatch results in poor performance and port errors. Remember, if you set speed and duplex on one side, you must set speed and duplex on the connecting device as well to avoid these problems.

In order to view the port status after setting the speed and duplex do **show port**.

```
Switch-A (enable) set port speed 2/1 100
Port(s) 2/1 speed set to 100Mbps.
Switch-A (enable) set port duplex 2/1 full
Port(s) 2/1 set to full-duplex.
Switch-A (enable) show port
```

Port	Name	Status	Vlan	Level	Duplex	Speed	Type
2/1		connected	1	normal	full	100	10/100BaseTX
...							

These are the timing results:

```
Switch-A (enable) show time
Fri Feb 25 2000, 14:05:28 Eastern
Switch-A (enable) set port enable 2/1
Port 2/1 enabled.
Switch-A (enable)
```

```
2000 Feb 25 140529 Eastern -0500 %PAGP-5-PORTTOSTP:Port 2/1 joined bridgeport
2/1
2000 Feb 25 140530 Eastern -0500 %SPANTREE-6-PORTFWD: port 2/1 state in vlan 1
changed to forwarding.
```

The final result gives a time of **2 seconds**(0528 to 0530).

You did another visually timed test by starting a continuous ping (ping -t) directed to the switch on a PC attached to the switch. you then disconnected the cable from the switch. The pings began to fail. Then you reconnected the cable to the switch and checked these watches to see how long it took for the switch to respond to the pings from the PC. It took about 5-6 seconds with auto-negotiation for speed and duplex turned on and about 4 seconds with auto-negotiation for speed and duplex turned off.

There are a lot of variables in this test (PC initialization, PC software, Switch console port responses to requests, and so on), but you just wanted to get some feel for how long it would take to get a response from the PCs point of view. All the tests were from the internal debug message point of view of the switches.

How to Reduce Startup Delay on the Catalyst 2900XL/3500XL Switch

The 2900XL and 3500XL models can be configured from a web browser, or by SNMP, or by the command line interface (CLI). you use the CLI. This is an example where you view the spanning tree state of a port, turn on portfast, and then verify that it is on. The 2900XL/3500XL does support EtherChannel and trunking, but it does not support dynamic EtherChannel creation (PAgP) or dynamic trunk negotiation (DTP) in the version you tested (11.2(8.2)SA6), so you have no need to turn them off in this test. Also, after you turn on portfast, the elapsed time for the port to come up is already less than 1 second, so there is not much point to try to change speed/duplex negotiation settings to speed things up. you hope one second is fast enough! By default, portfast is off on the switch ports. These are the commands to turn portfast on:

Configuration

```
2900XL#conf t
2900XL(config)#interface fastEthernet 0/1
2900XL(config-if)#spanning-tree portfast
2900XL(config-if)#exit
2900XL(config)#exit
2900XL#copy run start
```

This platform is like the router Cisco IOS; you must save the configuration (**copy run start**) if you want it to be permanently saved.

Verification

In order to verify that Portfast is enabled, do this command:

```
2900XL#show spanning-tree interface fastEthernet 0/1
Interface Fa0/1 (port 13) in Spanning tree 1 is FORWARDING
```

```
Port path cost 19, Port priority 128
Designated root has priority 8192, address 0010.0db1.7800
Designated bridge has priority 32768, address 0050.8039.ec40
Designated port is 13, path cost 19
Timers: message age 0, forward delay 0, hold 0
BPDU: sent 2105, received 1
The port is in the portfast mode
```

Look at the switch configuration.

```
2900XL#show running-config
Building configuration...

Current configuration:
!
version 11.2
...
!
interface VLAN1
 ip address 172.16.84.5 255.255.255.0
 no ip route-cache
!
interface FastEthernet0/1
 spanning-tree portfast
!
interface FastEthernet0/2
!
...
```

Timing Tests on the Catalyst 2900XL

These are the timing tests on the Catalyst 2900XL.

The 11.2(8.2)SA6 version of software was used on the 2900XL for these tests.

```
Switch#show version
Cisco Internetwork Operating System Software
Cisco IOS (tm) C2900XL Software (C2900XL-C3H2S-M), Version 11.2(8.2)SA6,
MAINTENANCE INTERIM SOFTWARE
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Wed 23-Jun-99 16:25 by boba
Image text-base: 0x00003000, data-base: 0x00259AEC

ROM: Bootstrap program is C2900XL boot loader

Switch uptime is 1 week, 4 days, 22 hours, 5 minutes
System restarted by power-on
System image file is "flash:c2900XL-c3h2s-mz-112.8.2-SA6.bin", booted via
console
```

cisco WS-C2924-XL (PowerPC403GA) processor (revision 0x11) with 8192K/1024K bytes of memory.

Processor board ID 0x0E, with hardware revision 0x01

Last reset from power-on

Processor is running Enterprise Edition Software

Cluster command switch capable

Cluster member switch capable

24 Ethernet/IEEE 802.3 interface(s)

32K bytes of flash-simulated non-volatile configuration memory.

Base ethernet MAC Address: 00:50:80:39:EC:40

Motherboard assembly number: 73-3382-04

Power supply part number: 34-0834-01

Motherboard serial number: FAA02499G7X

Model number: WS-C2924-XL-EN

System serial number: FAA0250U03P

Configuration register is 0xF

you want the switch to tell us what happens and when it happens, so you enter these commands:

```
2900XL(config)#service timestamps debug uptime
```

```
2900XL(config)#service timestamps log uptime
```

```
2900XL#debug span-tree events
```

```
Spanning Tree event debugging is on
```

```
2900XL#show debug
```

```
General spanning tree:
```

```
Spanning Tree event debugging is on
```

Then, you shut down the port in question.

```
2900XL#conf t
```

```
Enter configuration commands, one per line. End with CNTL/Z.
```

```
2900XL(config)#interface fastEthernet 0/1
```

```
2900XL(config-if)#shut
```

```
2900XL(config-if)#
```

```
00:31:28: ST: sent Topology Change Notice on FastEthernet0/6
```

```
00:31:28: ST: FastEthernet0/1 - blocking
```

```
00:31:28: %LINK-5-CHANGED: Interface FastEthernet0/1, changed state to administratively down
```

```
00:31:28: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down
```

```
2900XL(config-if)#exit
```

```
2900XL(config)#exit
```

```
2900XL#
```

At this point you paste these commands from the clipboard into the switch. These commands show the time on the 2900XL and turn the port back on:

```
show clock
conf t
int f0/1
no shut
```

By default, Portfast is off. You can confirm it two ways. The first way is that the **show spanning-tree interface** command does not mention Portfast. The second way is to look at this config that runs and where you do not see the **spanning-tree portfast** command under the interface.

```
2900XL#show spanning-tree interface fastEthernet 0/1
Interface Fa0/1 (port 13) in Spanning tree 1 is FORWARDING
  Port path cost 19, Port priority 128
  Designated root has priority 8192, address 0010.0db1.7800
  Designated bridge has priority 32768, address 0050.8039.ec40
  Designated port is 13, path cost 19
  Timers: message age 0, forward delay 0, hold 0
  BPDU: sent 887, received 1
[Note: there is no message about being in portfast mode is in this spot...]

2900XL#show running-config
Building configuration...
...
!
interface FastEthernet0/1
[Note: there is no spanning-tree portfast command under this interface...]
!
```

Here is the first timing test with Portfast off.

```
2900XL#show clock
*00:27:27.632 UTC Mon Mar 1 1993
2900XL#conf t
Enter configuration commands, one per line. End with CNTL/Z.
2900XL(config)#int f0/1
2900XL(config-if)#no shut
2900XL(config-if)#
00:27:27: ST: FastEthernet0/1 - listening
00:27:27: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
00:27:28: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1,
changed state to up
00:27:42: ST: FastEthernet0/1 - learning
00:27:57: ST: sent Topology Change Notice on FastEthernet0/6
```

00:27:57: ST: FastEthernet0/1 - **forwarding**

Total time from shutdown until the port started forwarding was **30 seconds**(27:27 to 27:57)

In order to turn on Portfast, do this:

```
2900XL#conf t
Enter configuration commands, one per line. End with CNTL/Z.
2900XL(config)#interface fastEthernet 0/1
2900XL(config-if)#spanning-tree portfast
2900XL(config-if)#exit
2900XL(config)#exit
2900XL#
```

In order to verify that Portfast is enabled, use the **show spanning-tree interface** command. Notice that the command output (near the end) indicates that Portfast is enabled.

```
2900XL#show spanning-tree interface fastEthernet 0/1
Interface Fa0/1 (port 13) in Spanning tree 1 is FORWARDING
  Port path cost 19, Port priority 128
  Designated root has priority 8192, address 0010.0db1.7800
  Designated bridge has priority 32768, address 0050.8039.ec40
  Designated port is 13, path cost 19
  Timers: message age 0, forward delay 0, hold 0
  BPDU: sent 1001, received 1
```

The port is in the portfast mode

You can also see that Portfast is enabled in the configuration output.

```
2900XL#sh ru
Building configuration...
...
interface FastEthernet0/1
  spanning-tree portfast
...
```

Now do the timing test with Portfast enabled

```
2900XL#show clock
*00:23:45.139 UTC Mon Mar 1 1993
2900XL#conf t
```

```
Enter configuration commands, one per line. End with CNTL/Z.
2900XL(config)#int f0/1
2900XL(config-if)#no shut
2900XL(config-if)#
00:23:45: ST: FastEthernet0/1 -jump to forwarding from blocking
00:23:45: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
00:23:45: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1,
changed state to up
```

In this case the total time was under **1 second**. If port initialization delay on the switch was the problem, portfast must solve it.

Remember, the switch does not currently support trunk negotiation, so you do not need to turn it off. Nor does it support PAgP for trunking, so you do not need to turn it off, either. The switch does support auto-negotiation of speed and duplex, but since the delay is so small this would not be a reason to turn it off.

you also did the ping test from a workstation to the switch. It took about 5-6 seconds for the response to come from the switch, whether auto-negotiation for speed and duplex was on or off.

How to Reduce Startup Delay on the Catalyst 1900/2800 Switch

The 1900/2820 refer to Portfast by another name: Spantree Start-Forwarding. For the version of software, you run (V8.01.05), the switches default to this: Portfast is enabled on the Ethernet (10Mbps) ports, and Portfast is disabled on the fast Ethernet (uplink) ports. So, when you **show run** to view the configuration, if an Ethernet port says nothing about Portfast, then Portfast is enabled. If it says "no spantree start-forwarding" in the configuration, Portfast is disabled. On a FastEthernet (100Mbps) port, the opposite is true: For a FastEthernet port, Portfast is only on if the port shows "spantree start-forwarding" in the configuration.

Here is an example of setting Portfast on a FastEthernet port. These examples use Enterprise edition software, Version 8. The 1900 automatically saves the configuration after changes have been made. Remember, you would not want Portfast enabled on any port that connects to another switch or hub, only if the port attaches to an end-station. The configuration is saved automatically to NVRAM.

Configuration

```
1900#show version
Cisco Catalyst 1900/2820 Enterprise Edition Software
Version V8.01.05
Copyright (c) Cisco Systems, Inc. 1993-1998
1900 uptime is 0day(s) 01hour(s) 10minute(s) 42second(s)
cisco Catalyst 1900 (486sxl) processor with 2048K/1024K bytes of memory
Hardware board revision is 5
Upgrade Status: No upgrade currently in progress.
Config File Status: No configuration upload/download is in progress
27 Fixed Ethernet/IEEE 802.3 interface(s)
```

```

Base Ethernet Address: 00-50-50-E1-A4-80
1900#conf t
Enter configuration commands, one per line.  End with CNTL/Z
1900(config)#interface FastEthernet 0/26
1900(config-if)#spantree start-forwarding
1900(config-if)#exit
1900(config)#exit
1900#

```

Verification

One way to verify that portfast is on is to look at the configuration. Remember, a FastEthernet port must say it is on. An Ethernet port has it on unless the configuration shows that it is off. In this configuration, interface Ethernet 0/1 has portfast turned off (you can see the command to turn it off), interface Ethernet 0/2 has portfast on (you see nothing - which means it is on), and interface FastEthernet 0/26 (port A in the menu system) has portfast on (you can see the command to turn it on).

```

1900#show running-config
Building configuration...
...
!
interface Ethernet 0/1

    no spantree start-forwarding
!
interface Ethernet 0/2

!
...
!
interface FastEthernet 0/26
    spantree start-forwarding

```

The easiest way to view the portfast status is through the menu system. If you choose **(P)** for Port Configuration from the main menu, then choose **aport**, the output indicates whether Port fast mode is enabled. This output is for port FastEthernet 0/26, which is port "A" on this switch.

Catalyst 1900 - Port A Configuration

```

Built-in 100Base-FX
802.1d STP State:  Blocking      Forward Transitions:  0

```

```

----- Settings -----
[D] Description/name of port
[S] Status of port                Suspended-no-linkbeat
[I] Port priority (spanning tree) 128 (80 hex)
[C] Path cost (spanning tree)     10
[H] Port fast mode (spanning tree) Enabled
[E] Enhanced congestion control    Disabled

```

[F] Full duplex / Flow control

Half-Duplex

----- Related Menu -----

[A] Port addressing

[V] View port statistics

[N] Next port

[G] Goto port

[P] Previous port

[X] Exit to Main Menu

Enter Selection:

Timing Tests on the Catalyst 1900

The timing values are harder to verify on a 1900/2820 because of the lack of debugging tools, so you just started a ping from a PC connected to the switch directed to the switch itself. you disconnected and then reconnected the cable and recorded how long it took for the switch to respond to the ping with Portfast on and with Portfast off. For an Ethernet port with Portfast on (the default state), the PC received a response within **5-6 seconds**. With Portfast off the PC received a response in 34-35 seconds.

An Additional Benefit to Portfast

There is another spanning tree-related benefit to the use of Portfast in your network. Every time a link becomes active and moves to the forwarding state in spanning tree, the switch sends a special spanning tree packet called a Topology Change Notification (TCN). The TCN notification is passed up to the root of the spanning tree, where it is propagated to all the switches in the VLAN. This causes all the switches to age out their table of MAC addresses with the forward delay parameter. The forward delay parameter is usually set to 15 seconds. Every time a workstation joins the bridge group, the MAC addresses on all the switches are aged out after 15 seconds instead of the normal 300 seconds.

Since when a workstation becomes active it does not really change the topology to any significant degree as far as all the switches in the VLAN are concerned, it is unnecessary for them to have to go through the fast aging TCN period. If you turn on portfast, the switch does not send TCN packets when a port becomes active.

Commands to Use to Verify the Configuration Works

This is a list of commands to use when you verify whether the configuration works.

4000/5000/6000

show port spantree 2/1- see if "Fast-Start" (Portfast) is enabled or disabled

show spantree 1- see all ports in VLAN 1 and if they have "Fast-Start" enabled

show port channel- see if you have any active channels

show port channel 2- see the channel mode (auto, off, and so on) for each port on module 2

show trunk 2- see the trunk mode (auto, off, and so on) for each port on module 2

show port- see the status (connected, notconnect, and so on), speed, duplex for all ports on the switch

2900XL/3500XL

show spanning-tree interface FastEthernet 0/1- to see if Portfast is enabled on this port (no mention of Portfast means that it is not enabled)

show running-config- if a port shows the command spanning-tree portfast then Portfast is enabled

1900/2800

show running-config- to see the current settings (some commands are invisible when they represent the default settings of the switch)

Use the menu system to the port status screen

Commands to Use to Troubleshoot the Configuration

This is a list of commands to use to troubleshoot the configuration.

4000/5000/6000

show port spantree 2/1- see if "Fast-Start" (Portfast) is enabled or disabled

show spantree 1- see all ports in VLAN 1 and if they have "Fast-Start" enabled

show port channel- see if you have any active channels

show port channel 2- see the channel mode (auto, off, and so on) for each port on module 2

show trunk 2- see the trunk mode (auto, off, and so on) for each port on module 2

show port- see the status (connected, notconnect, and do on), speed, duplex for all ports on the switch

show logging- see what type of messages generate logging output

set logging level spantree 7- sets the switch to log the spanning tree port, states real time on the console

set port disable 2/1- turn the port off in software (like "shutdown" on the router)

set port enable 2/1- turn the port on in software (like "no shutdown" on the router)

show time- show the current time in seconds (used at the start of a timing test)

show port capabilities- see what features are implemented on the port

set trunk 2/1 off- set the trunking mode to off (to speed port initialization time)

set port channel 2/1-2 off- set the EtherChannel (PAgP) mode to off (to speed port initialization time)

set port speed 2/1 100- set the port to 100Mbps and turn off auto-negotiation

set port duplex 2/1 full- set the port duplex to full

2900XL/3500XL

service timestamps debug uptime- show the time with the debug messages

service timestamps log uptime- show the time with the logging messages

debug spantree events- show when the port moves through the spanning tree stages

show clock- to see the current time (for the timing tests)

show spanning-tree interface FastEthernet 0/1- to see if Portfast is enabled on this port (no mention of Portfast means that it is not enabled)

shut- to turn off a port from software

no shut- to turn on a port from software

1900/2800

show running-config- to see the current settings (some commands are invisible when they represent the default settings of the switch)

Configure and Troubleshoot IP Multilayer Switching (MLS)

Objectives

This document outlines how to troubleshoot Multilayer Switching (MLS) for IP. This feature has become a highly desired method with which to accelerate routing performance through the use of dedicated Application Specific Integrated Circuits (ASICs). Traditional routing is done through a central CPU and software; MLS offloads a significant portion of routing (packet rewrite) to hardware and has also been termed switching. MLS and layer-three switching are equivalent terms. The NetFlow feature of Cisco IOS is distinct, and not covered in this document. MLS also includes support for IPX (IPX MLS) and multicasting (MPLS), but this document exclusively concentrates on how to troubleshoot basic MLS IP.

Introduction

As greater demands are placed on networks, the need for greater performance increases. More and more PCs are connected to LANs, WANs and the Internet, and their users require fast access to databases, files/webpages, networked applications, other PCs, and streaming video. In order to keep connections quick and reliable, networks must be able to rapidly adjust to changes and failures and find the best path, all while they remain as invisible as possible to end users. End users that experience rapid information flow between their PC and server with minimal network slowness are happy ones. Determination of the best path is the primary function of routing protocols, and this can be a CPU-intensive process; a significant performance increase is gained by offloading a portion of this function to switching hardware. This is the point of the MLS feature.

There are three major components of MLS: two of them are the MLS-RP and the MLS-SE. The MLS-RP is the MLS-enabled router, which performs the traditional function of routing between subnets/VLANs. The MLS-SE is an MLS-enabled switch, which normally requires a router to route between subnets/VLANs, but with special hardware and software, can handle rewriting of the packet. When a packet transverses a routed interface, non-data portions of the packet are changed (rewritten) as it is carried to its destination, hop by hop. Confusion can arise here, since it seems that a layer-two device takes on a layer-three task; actually, the switch is only rewriting layer-three information, and is switching between subnets/VLANs--the router is still responsible for standards-based route calculations and best-path determination. Much of this confusion can be avoided if you mentally keep the routing and switching functions separate, especially when, as is commonly the case, they are contained within the same chassis (as with an internal MLS-RP). Think of MLS as a much more advanced way to cache the router, with the cache kept separate from the router on a switch. Both the MLS-RP and the MLS-SE, along with respective hardware and software minimums, are required for MLS.

The MLS-RP can be internal (installed in a switch chassis) or external (connected through a cable to a trunk port on the switch). Examples of internal MLS-RPs are the Route-Switch Module (RSM) and the Route-Switch Feature Card (RSFC), which are installed in a slot or supervisor of a Catalyst 5xxx family member, respectively; the same applies to the Multilayer Switch Feature Card (MSFC) for the Catalyst 6xxx family. Examples of external MLS-RPs include any member of the Cisco 7500, 7200, 4700, 4500 or 3600 series routers. In general, to support the MLS IP feature, all MLS-RPs require a minimum Cisco IOS version in the 11.3WA or 12.0WA trains; consult release documentation for specifics. Also, **MLS must be enabled** in order for a router to be an MLS-RP.

The MLS-SE is a switch with special hardware. For a member of the Catalyst 5xxx family, MLS requires that the supervisor have a NetFlow Feature Card (NFFC) installed; the Supervisor IIG and IIIG have one by default. In addition, a bare minimum of Catalyst OS 4.1.1 software is also required. Note that the 4.x train has 'gone General Deployment (GD),' or passed rigorous end-user criteria and field-experience targets for stability, so check the Cisco website for the latest releases. IP MLS is supported and automatically enabled for Catalyst 6xxx hardware and software with the

MSFC/PFC (other routers have MLS disabled by default). Note that IPX MLS and MLS for multicasting can have different hardware and software (Cisco IOS and Catalyst OS) requirements. More Cisco platforms do/can support the MLS feature. Also, **MLS must be enabled** in order for a switch to be an MLS-SE.

The third major component of MLS is the Multilayer Switching Protocol (MLSP). This is because when you understand the basics of MLSP you get at the heart of MLS, and this is essential to effectively troubleshoot the MLS. MLSP is utilized by the MLS-RP and the MLS-SE to communicate with one another; tasks that enable MLS, and install, update, or delete flows (cache information), and the management and export of the flow statistics (NetFlow Data Export is covered in other documentation). MLSP also allows the MLS-SE to learn the Media Access Control (MAC, layer-two) addresses of the MLS-enabled router interfaces, check the flowmask of the MLS-RP (explained later in this document), and confirm that the MLS-RP is operational. The MLS-RP sends out multicast 'hello' packets every 15 seconds with MLSP; if three of these intervals are missed, then the MLS-SE recognizes that the MLS-RP has failed or that connectivity to it has been lost.

The diagram illustrates three essentials that must be completed (with MLSP) for a shortcut to be created: the candidate, enabler and caching steps. The MLS-SE checks for a cached MLS entry; if MLS cache entry and packet information match (a hit), the header of the packet is rewritten locally on the switch (a shortcut or bypass of the router) instead of sent on to the router as normally happens. Packets that do not match and are sent on to the MLS-RP are candidate packets; that is, there is a possibility of switching them locally. After it passes the candidate packet through the MLS flowmask (explained in a section later) and rewrites the information contained in the header of the packet (the data portion is not touched), the router sends it toward the next hop along the destination path. The packet is now called an enabler packet. If the packet returns to the same MLS-SE from which it left, an MLS shortcut is created and placed into the MLS cache; rewriting for that packet and all similar packets that trace them (called a flow) is now done locally by switch hardware instead of by router software. **The same MLS-SE must see both the candidate and enabler packets for a particular flow for an MLS shortcut to be created** (this is why network topology is important to MLS). Remember, the point of MLS is to allow the communication path between two devices in different VLANs, connected off of the same switch, to bypass the router, and enhance network performance.

By the use of the flowmask (essentially an access list) the administrator can adjust the degree of similarity of these packets and adjust the scope of the flows: destination address; destination and source addresses; or destination, source and layer-four information. Note that the first packet of a flow always passes through the router; from then on it is locally switched. Each flow is unidirectional; communication between PCs, for example, requires the setup and use of two shortcuts. The main purpose of MLSP is to setup, create, and maintain these shortcuts.

These three components (the MLS-RP, the MLS-SE and MLSP) free up vital router resources when it allows other network components to take on some of its functions. Dependent upon the topology and configuration, MLS provides a simple and highly effective method that increases network performance in the LAN.

Troubleshoot IP MLS Technology

A flow diagram to use to troubleshoot basic IP MLS is included and discussed. It is derived from the most common types of MLS-IP cases opened with the Cisco Technical Support Website and faced by the users and Technical Support engineers up to the time that this document was created. MLS is a robust feature, and you must have no problems with it; if an issue does arise, this helps you to resolve the types of IP MLS problems you most likely face. A few essential assumptions are made:

You are familiar with the basic configuration steps required to enable IP MLS on the router and switches and have completed these steps: see the resources listed at the end of this document for excellent material.

The IP routing is enabled on the MLS-RP (it is on by default): if the command **no ip routing** appears in the global configuration of **show run**, it has been turned off, and IP MLS does not function.

IP connectivity exists between the MLS-RP and MLS-SE: **ping** the IP addresses of the router from the switch and look for exclamation points (called 'bangs') to display in return.

The MLS-RP interfaces are in an 'up/up' state on the router: type **show ip interface brief** on the router to confirm this.

Warning: Whenever you make configuration changes to a router intended to be permanent, remember to save those changes with **copy running-config starting-config** (shortened versions of this command include **copy run start** and **wr mem**). Any configuration modifications are lost if the router reloads or is reset. The RSM, RSFC and MSFC are routers, not switches. In contrast, changes made at the switch prompt of a Catalyst 5xxx or 6xxx family member are automatically saved.

This section troubleshoots IP MLS technology.

Are minimum hardware and software requirements met?

Upgrade the MLS-RP and SE to meet minimum software and hardware requirements. For the MLS-RP, no additional hardware is required. Although MLS can be configured on non-trunked interfaces, the connection to the MLS-SE is generally through VLAN interfaces (as with an RSM) or support trunking (can be configured to carry multiple VLAN information by configuring ISL or 802.1q). Also, remember that, as of publication time, only members of the 7500, 7200, 4700, 4500, and 3600 router families support MLS externally. Currently, only these external routers and the routers which fit into the Catalyst 5xxx or 6xxx switch families (like the RSM and RSFC for the Catalyst 5xxx family, and the MSFC for the Catalyst 6xxx family) can be MLS-RPs. The MSFC requires the Policy Feature Card (PFC) as well, both installed on the Catalyst 6xx Supervisor. IP MLS is now a standard feature in Cisco IOS 12.0 and later router software. Cisco IOS software lower than Cisco IOS 12.0 generally requires a special train; for such IP MLS support, install the latest images in Cisco IOS 11.3 that have the letters 'WA' in their filenames.

For the MLS-SE, a NetFlow Feature Card (NFFC) is required for a member of the Catalyst 5xxx family; this card is installed in the Supervisor module of the Catalyst switch and is included as standard hardware in newer Catalyst 5xxx series Supervisors (that is, since 1999). The NFFC is not supported on the Supervisors I or II and is an option on early Supervisor IIIs. Also, a minimum of 4.1.1 CatOS is required for IP MLS. In contrast, for the Catalyst 6xxx family, the required hardware comes as standard equipment, and IP MLS has been supported since the first CatOS software release, 5.1.1 (in fact, IP MLS is an essential and default ingredient for its high performance). With new platforms and software released that support IP MLS, it is important to check documentation and release notes, and to generally install the latest release in the lowest train that meets your feature requirements. Always check the release notes and consult with your local Cisco sales office for new MLS

support and feature developments.

Commands used to check the installed hardware and software are **show version** on the router and **show module** on the switch

Note: The Catalyst 6xxx family of switches does NOT support an external MLS-RP at this time. The MLS-RP must be an MSFC.

Are the source and destination devices in different VLANs off the same MLS-SE, sharing a single common MLS-RP?

It is a basic topology requirement of MLS that the router have a path to each of the VLANs. Remember that the point of MLS is to create a shortcut between two VLANs, so that the routing between the two end devices can be performed by the switch, and this frees the router for other tasks. The switch does not actually route; it rewrites the frames so that it appears to the end devices that they talk through the router. If the two devices are in the same VLAN, the MLS-SE switches the frame locally without the use of MLS, as switches do in such a transparently bridged environment, and no MLS shortcut is created. One can have multiple switches and routers in the network, and even multiple switches along the flow path, but the path between the two end devices for which one desires an MLS shortcut must include a single MLS-RP in that VLAN for that path. In another words, the flow from source to destination must cross a VLAN boundary on the same MLS-RP, and a candidate and enabler packet pair must be seen by the same MLS-SE for the MLS shortcut to be created. If these criteria are not met, the packet is routed normally without the use of MLS. Refer to the documents suggested at the end of this document for diagrams and discussions in regard to supported and unsupported network topologies.

Does the MLS-RP contain an **mpls rp ip** statement under both its global and interface configuration?

If one is not present, add **mpls rp ip** statements appropriately on the MLS-RP. Except for routers for which IP MLS is automatically enabled (like the Catalyst 6xxx MSFC), this is a required configuration step. For most MLS-RPs (routers configured for IP MLS), this statement must appear both in the global configuration and under the interface configuration.

Note: When you configure the MLS-RP, also remember to place the **mpls rp management-interface** command under one of its IP MLS interfaces. This required step tells the MLS-RP out of which interface it must send MLSP messages to communicate with the MLS-SE. Again, it is necessary to place this command under one interface only.

Are there any features configured on the MLS-RP that automatically disable MLS on that interface?

There are several configuration options on the router which are not compatible with MLS. These include IP accounting, encryption, compression, IP security, Network Address Translation (NAT), and Committed Access Rate (CAR). For further information, refer to links in regard to IP MLS configuration included at the end of this document. Packets that traverse a router interface configured with any of these features must be routed normally; no MLS shortcut are created. For MLS to work, disable these features on the MLS-RP interface.

Another important feature that affects MLS is access lists, both input and output. Further discussion of this option is included under 'flowmasks.'

Does the MLS-SE recognize the MLS-RP address?

For MLS to function, the switch must recognize the router as an MLS-RP. Internal MLS-RPs (once again, the RSM or RSFC in a Catalyst 5xxx family member, and the MSFC in a Catalyst 6xxx family member) are automatically recognized by the MLS-SE in which they are installed. For external MLS-RPs, one must explicitly inform the switch of the address of the router. This address is not actually an IP address, although on external MLS-RPs it is chosen from the list of IP addresses configured on the interfaces of the router; it is simply a router ID. In fact, for internal MLS-RPs, the MLS-ID is normally not even an IP address configured on the router; since internal MLS-RPs are included automatically, it is commonly a loopback address (127.0.0.x). For MLS to function, include on the MLS-SE the MLS-ID found on the MLS-RP.

Use **show mls rp** on the router to find the MLS-ID. Then configure that ID on the switch with the **set mls include <MLS-ID>** command. This is a required configuration step when you use external MLS-RPs.

Note: If you change the IP address of MLS-RP interfaces and then reload the router, it can cause the MLS process on the router to choose a new MLS-ID. This new MLS-ID can be different from the MLS-ID that was manually included on the MLS-SE, which can cause MLS to stop; this is not a software glitch, but an effect of the switch that tries to communicate with an MLS-ID that is no longer valid. Be sure to include this new MLS-ID on the switch for the MLS to work once again. It can be necessary to disable/enable IP MLS, as well.

Note: When the MLS-SE is not directly connected to the MLS-RP, as with this topology, the address that must be included on the MLS-SE can appear as the loopback address mentioned: a switch connected in between the MLS-SE and MLS-RP. You must include the MLS-ID even though the MLS-RP is internal. To the second switch, the MLS-RP appears as an external router since the MLS-RP and MLS-SE are not contained in the same chassis.

Are the MLS-RP interface and the MLS-SE in the same enabled VTP domain?

MLS requires that MLS components, along with the end stations, must be in the same Virtual Trunking Protocol (VTP) domain. VTP is a layer-two protocol used for managing VLANs on several Catalyst switches from a central switch. It allows an administrator to create or delete a VLAN on all switches in a domain and not have to do so on every switch in that domain. The Multilayer Switching Protocol (MLSP), which the MLS-SE and the MLS-RP use to communicate with one another, does not cross a VTP domain boundary. If the network administrator has VTP enabled on the switches (VTP is enabled on Catalyst 5xxx and 6xxx family members by default), use the **show vtp domain** command on the switch to learn in which VTP domain the MLS-SE has been placed. Except for the Catalyst 6xxx MSFC, on which MLS is essentially *a plug-and-play* feature, you need to next add the VTP domain to each of the MLS interfaces of the router. This permits MLSP multicasts to move between the MLS-RP and MLS-SE and allows MLS to function.

In interface configuration mode of the MLS-RP, enter these commands:

no mls rp ip Disable MLS on the affected MLS-RP interface before you modify the VTP domain.

mls rp vtp-domain < VTP domain name > VTP domain name on each MLS-enabled interface must match that of the switch.

mls rp vlan-id <VLAN # > Only required for non-ISL trunking, external MLS-RP interfaces.

mls rp management-interface Do this for only one interface on the MLS-RP. This required step tells the MLS-RP out of which interface it must send MLSP messages.

mls rp ip Enable MLS once again on the interface of the MLS-RP.

In order to change the VTP domain name of the MLS-SE, use this command at the switch CatOS enable prompt:

set vtp domain name <VTP domain name>

For MLS to work, be sure that VTP is enabled on the switch:

set vtp enable

Do the flowmasks agree on the MLS-RP and MLS-SE?

A flowmask is a filter configured by a network administrator that is used by MLS to determine whether a shortcut needs to be created. Just like an access list, the more detailed the criteria you set up, the deeper into the packet the MLS process must look in order to verify if the packet meets those criteria. In order to adjust the scope of MLS-created shortcuts, the flowmask can be made more or less specific; the flowmask is essentially *atuning* device. There are three types of IP MLS modes: destination-IP, destination-source-IP, and full-flow-IP. Destination-IP mode, the default, is in use when no access list is applied to the MLS-enabled interface of the router. Source-destination-IP mode is in use when a standard access list is applied. Full-flow-IP is in effect for an extended access list. The MLS mode on the MLS-RP is implicitly determined by the type of access list applied to the interface. In contrast, the MLS mode on the MLS-SE is explicitly configured. If the appropriate mode is chosen, the user can thus configure MLS so that only the destination address must match in order for an MLS shortcut to be created, or both source and destination, or even layer-four information like TCP/UDP port numbers.

The MLS mode is configurable on both the MLS-RP and the MLS-SE, and in general, they must match. IF either source-destination-IP or full-flow-IP MLS modes are deemed to be required, it is best to configure it on the router by and apply the appropriate access list. MLS always chooses the most specific mask. It gives the flowmask configured on the MLS-RP precedence over the one found on the MLS-SE. BE CAREFUL if you change the MLS mode of the switch from the default destination-ip: you must make sure that it matches the MLS mode on the router for MLS to work. For source-destination-ip and full-flow-ip modes, remember to apply the access list to the appropriate router interface; with no access list applied, even if configured, the MLS mode simply is destination-ip, the default.

Warning: Whenever the flowmask is changed, whether on the MLS-RP or MLS-SE, all cached MLS flows are purged, and the MLS process is restarted. A purge also can occur when you apply the command **clear ip route-cache** on the router. If you apply the global router configuration command **no ip routing**, which turns off IP routing and essentially transforms the router into a transparent bridge, it causes a purge and disable MLS (remember, routing is a prerequisite of MLS). Each of these can temporarily, but seriously, affect router performance in a production network. The router experiences a spike in its load until the new shortcuts are created because it must now handle all the flows that were previously processed by the switch.

Note: Especially with a member of the Catalyst 5000 family as the MLS-SE, you must avoid the very wide use of flowmasks that are configured with layer-four information. If the router is forced to peer so deeply into every packet on the interface, much of the intended benefits of

MLS are bypassed. This is much less of an issue when you use a Catalyst 6xxx family member as the MLS-SE since the switch ports themselves can recognize layer-four information.

Note: Until recently, MLS did not support flowmasks configured inbound on an MLS-RP interface, only outbound. If you use the **mls rp ip input-acl** command in addition to normal MLS-RP configuration commands on a router interface, an inbound flowmask is supported.

Are more than a couple of MLS *Too many moves* error messages continuously seen on the switch?

As the note mentions, to change a flowmask, clear the route cache, or globally turn off IP routing causes a cache purge. Other circumstances can also cause full or many single entry purges and cause MLS to complain of *Too many moves*. There are several forms of this message, but each contains these three words. Aside from what has already been mentioned, the most common cause of this error is when the switch learns multiple identical Ethernet Media Access Control (MAC) address within the same VLAN; Ethernet standards do not allow for identical MAC addresses within the same VLAN. If seen infrequently, or just a few times in a row, there is no cause for concern; MLS is a robust feature, and the message can be simply caused by normal network events, like a PC connection moved between ports, for example. If seen continuously for several minutes, it is likely a symptom of a more serious issue.

When such a situation arises, its root cause is commonly due to the presence of two devices with the same MAC address actually connected to a VLAN, or a physical loop within the VLAN (or multiple VLANs if bridging across these broadcast domains). Troubleshoot with spanning-tree (covered in other documents) and the tip to find the loop and eliminate it. Also, any rapid topology changes can cause temporary network (and MLS) instability (flapping router interfaces, a bad network interface card (NIC), and so on).

Tip: Use the **show mls notification** and **show looktable** commands on the switch to point you in the direction of the duplicate MAC address or physical loop. The first provides a TA value. The command **show looktable <TA value>** returns a possible MAC address that can be traced to the root of the problem.

Related Information

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Conventions](#)

[Background Information](#)

[LAN Switching Introduction](#)

[Hubs and Switches](#)

[Bridges and Switches](#)

[VLANs](#)

[Transparent Bridging Algorithm](#)

[Spanning Tree Protocol](#)

[Trunking](#)

[EtherChannel](#)

[Multilayer Switching \(MLS\)](#)

[How to Learn About These Features](#)

[Suggestion to Troubleshoot General Switch](#)

[Troubleshoot Port Connectivity Problems](#)

[Hardware Issues](#)

[Configuration Issues](#)

[Traffic Issues](#)

[Switch Hardware Failure](#)

[Troubleshoot Ethernet 10/100Mb Half/Full Duplex Auto-Negotiation](#)

[Objectives](#)

[Introduction](#)

[Troubleshoot Ethernet Auto-negotiation Between Network Infrastructure Devices](#)

[Procedures and/or Scenarios](#)

[Configure and Troubleshoot Ethernet 10/100Mb Auto-Negotiation Example](#)

[Step-by-Step](#)

[Before You Call the Cisco Systems Technical Support Team](#)

[Configure EtherChannel Switch-to-Switch Connections on Catalyst 4000/5000/6000 Switches](#)

[Tasks for Manual Configuration of EtherChannel](#)

[Step-by-Step](#)

[Verify the Configuration](#)

[Use PAgP to Configure EtherChannel \(Preferred Method\)](#)

[Trunking and EtherChannel](#)

[Troubleshoot EtherChannel](#)

[Commands Used in this Section](#)

[Use Portfast and Other Commands to Fix End-Station Startup Connectivity Problems](#)

[Contents](#)

[Background](#)

[How to Reduce Startup Delay on the Catalyst 4000/5000/6000 Switch](#)

[Timing Tests With and Without DTP, PAgP, and Portfast on a Catalyst 5000](#)

[How to Reduce Startup Delay on the Catalyst 2900XL/3500XL Switch](#)

[Timing Tests on the Catalyst 2900XL](#)

[How to Reduce Startup Delay on the Catalyst 1900/2800 Switch](#)

[Timing Tests on the Catalyst 1900](#)

[An Additional Benefit to Portfast](#)

[Commands to Use to Verify the Configuration Works](#)

[Commands to Use to Troubleshoot the Configuration](#)

[Configure and Troubleshoot IP Multilayer Switching \(MLS\)](#)

[Objectives](#)

[Introduction](#)

[Troubleshoot IP MLS Technology](#)

[Related Information](#)

- [Cisco Technical Support & Downloads](#)