AlliedWare Plus™ OS

Overview of | VLANs (Virtual LANs)

Introduction

This Overview describes VLANs—what they are, why they were developed, and how they behave.

It focuses on the general principles of VLANs, instead of describing any particular switch implementation.

Contents

What is a VLAN?

In simple terms, a VLAN is a set of workstations within a LAN that can communicate with each other as though they were on a single, isolated LAN.

What does it mean to say that they “communicate with each other as though they were on a single, isolated LAN”?

Among other things, it means that:

- broadcast packets sent by one of the workstations will reach **all** the others in the VLAN
- broadcasts sent by one of the workstations in the VLAN will not reach any workstations that are not in the VLAN
- broadcasts sent by workstations that are not in the VLAN will never reach workstations that are in the VLAN
- the workstations can all communicate with each other without needing to go through a gateway. For example, IP connections would be established by ARPing for the destination IP and sending packets directly to the destination workstation—there would be no need to send packets to the IP gateway to be forwarded on.
- the workstations can communicate with each other using non-routable protocols.

The purpose of VLANs

The basic reason for splitting a network into VLANs is to reduce congestion on a large LAN. To understand this problem, we need to look briefly at how LANs have developed over the years.

Initially LANs were very flat—all the workstations were connected to a single piece of coaxial cable, or to sets of chained hubs. In a flat LAN, every packet that any device puts onto the wire gets sent to **every** other device on the LAN.

As the number of workstations on the typical LAN grew, they started to become hopelessly congested; there were just too many collisions, because most of the time when a workstation tried to send a packet, it would find that the wire was already occupied by a packet sent by some other device.

This section describes the three solutions for this congestion that were developed:

- Using routers to segment LANs
- Using switches to segment LANs
- Using VLANs to segment LANs

Using routers to segment LANs

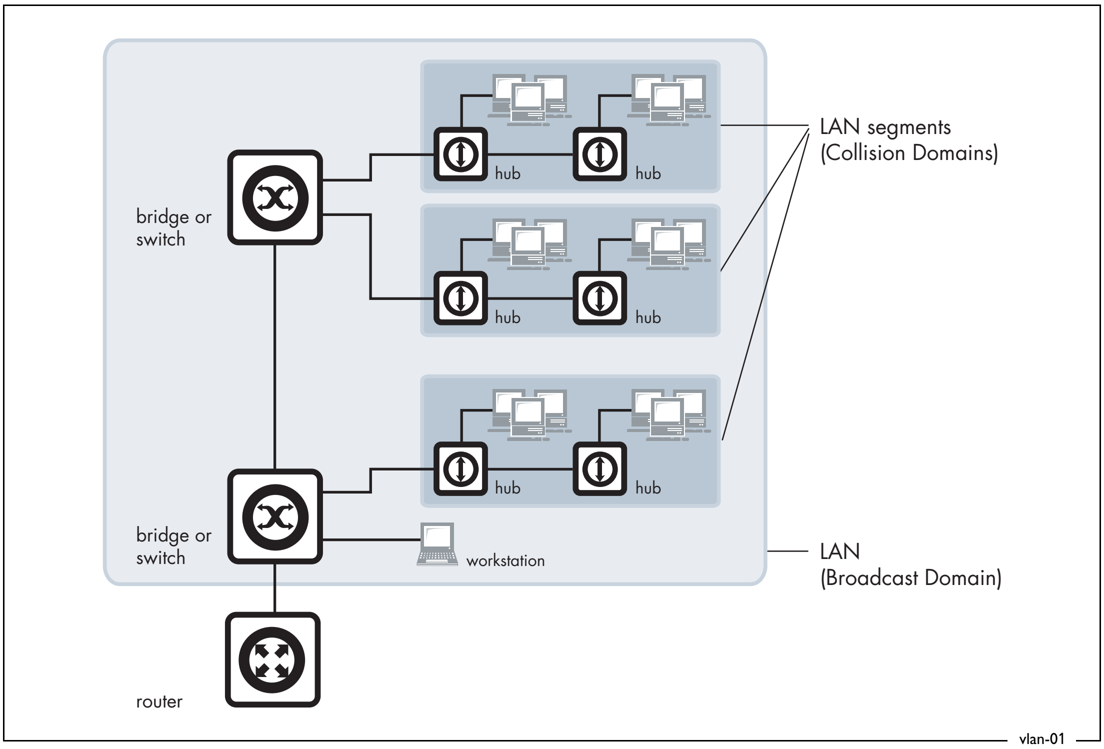
The early solution to this problem was to segment the network using routers. This would split the network into a number of smaller LANs. There would be less workstations on each LAN, and so less congestion.

Of course, routable data being sent between LANs would have to be routed, so the layer 3 addresses would have to be organized so that each LAN had an identifiable set of addresses that could be routed to—such as an IP subnet or an AppleTalk zone. Non-routable protocols would have to be bridged, which is not quite so congestion-reducing, because bridges forward all broadcasts. But, at least for unicast packets, a bridge only forwards packets if it knows that the destination address is not in the originating LAN.

Using switches to segment LANs

As switches became more available, there was a move from chained hubs to a set of hubs connected to a switch. A switch only sends traffic to a given port if the traffic **has** to go to that port. So switches have the effect of reducing congestion at workstations, by stopping the workstations from seeing all the traffic from the other ports of the switch.

A simple switched network, though, still needs routers to set the boundaries of where broadcasts are sent (referred to as "broadcast containment"). So, the typical LAN was set up as shown in the following figure:



Domain terminology

The above figure introduces the concept of a **LAN segment**. This is also referred to as a **collision domain**, because when a device is trying to send a packet, it can only collide with packets sent by other devices on the same segment. Each LAN segment consists of all the devices attached to a single switch port—the switch stops packets from different ports from colliding with each other.

The LAN itself is referred to as a **broadcast domain**, because if any device within the LAN sends out a broadcast packet, it will be transmitted to all devices in that LAN, but not to devices beyond the LAN.

Using VLANs to segment LANs

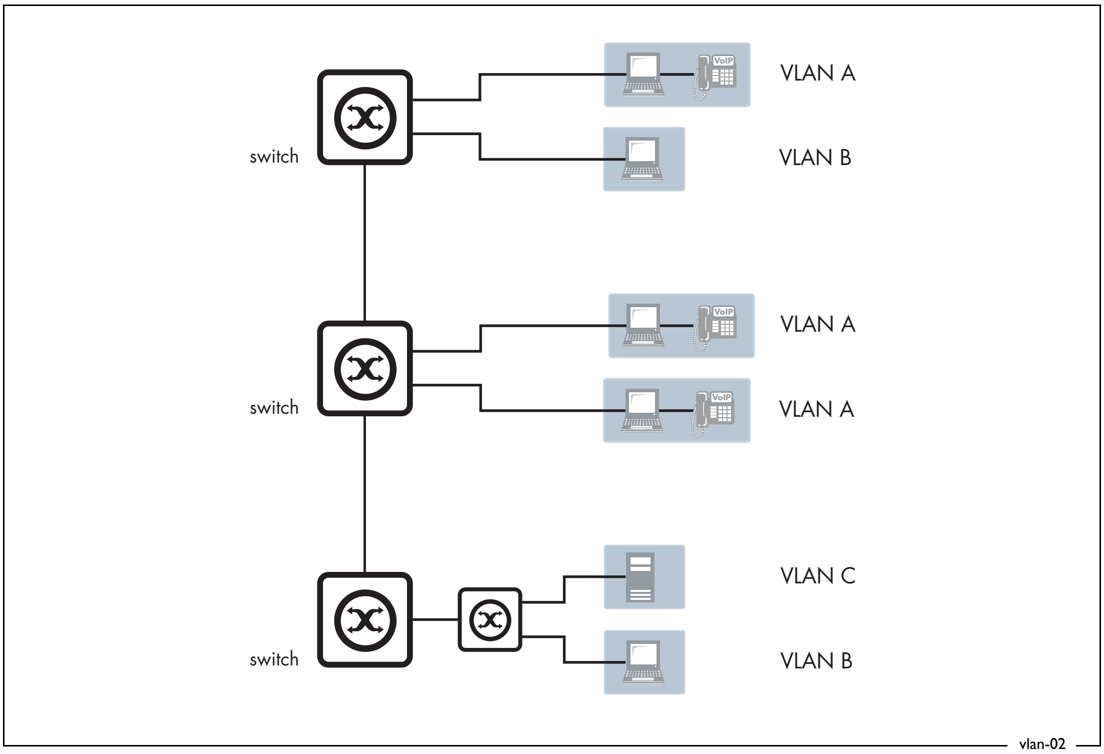
As LANs became larger, data rates became faster, and users desired greater flexibility, the routers in a network started to become a bottleneck. This is because:

- routers typically forward data in software, and so are not as fast as switches
- splitting up a LAN using routers meant that a LAN typically corresponded to a particular physical location. This became limiting when many users had laptops, and wanted to be able to move between buildings, but still have the same network environment wherever they plugged in.

So, switch vendors started implementing methods for defining "virtual LANs"—sets of switch ports, usually distributed across multiple switches, that somehow interacted as though they were in a single isolated LAN. This way, workstations could be separated off into separate LANs without being physically divided up by routers.

At about the same time, hubs became less popular and have been largely replaced by L2 switches. This has made the whole concept of a collision domain somewhat historical. In modern networks, a "collision domain" mostly consists of a single device attached to an L2 switch port, or possibly a PC with something like an IP phone attached to it.

So, the layout of the LAN has become more like:



So, instead of the LANs corresponding to physical areas divided from each other by routers, there are virtual LANs distributed across the network. For example, all the devices in the various areas labelled "VLAN A" all belong to a single virtual LAN—i.e. a single broadcast domain.

Advantages of using VLANs

1. **Performance**. As mentioned above, routers that forward data in software become a bottleneck as LAN data rates increase. Doing away with the routers removes this bottleneck.

2. **Formation of virtual workgroups**. Because workstations can be moved from one VLAN to another just by changing the configuration on switches, it is relatively easy to put all the people working together on a particular project all into a single VLAN. They can then more easily share files and resources with each other.

   To be honest, though, virtual workgroups sound like a good idea in theory, but often do not work well in practice. It turns out that users are usually more interested in accessing company-wide resources (file servers, printers, etc.) than files on each others' PCs.

3. **Greater flexibility**. If users move their desks, or just move around the place with their laptops, then, if the VLANs are set up the right way, they can plug their PC in at the new location, and still be within the same VLAN. This is much harder when a network is physically divided up by routers.

4. **Ease of partitioning off resources**. If there are servers or other equipment to which the network administrator wishes to limit access, then they can be put off into their own VLAN. Then users in other VLANs can be given access selectively.

Implementing VLANs

Port-based VLANs

In the previous section, we simply stated that the network is split up into sets of virtual LANs. It is one thing to say this; it is quite another thing to understand how this is actually achieved.

Fundamentally, the act of creating a VLAN on a switch involves defining a set of ports, and defining the criteria for VLAN membership for workstations connected to those ports.

By far the most common VLAN membership criterium is port-based. We will consider that criterium here, and visit the other options in "Other VLAN classification criteria" on page 10.

With port-based VLANs, the ports of a switch are simply assigned to VLANs, with no extra criteria.

Port	VLAN
1	1
2	1
3	2
4	1

All devices connected to a given port automatically become members of the VLAN to which that port was assigned.

In effect, this just divides a switch up into a set of independent sub-switches.

Distributing a single VLAN across multiple switches

The figure in "Using VLANs to segment LANs" on page 5 shows an example of a VLAN-based network. It shows some of VLAN A connected to one switch, and some more of VLAN A connected to another switch.

You may be asking “Are these both part of the same VLAN A, or separate VLANs that all happen to be called VLAN A?”

The answer is that they are all parts of the same VLAN—there is a single VLAN A that is spread across two switches.

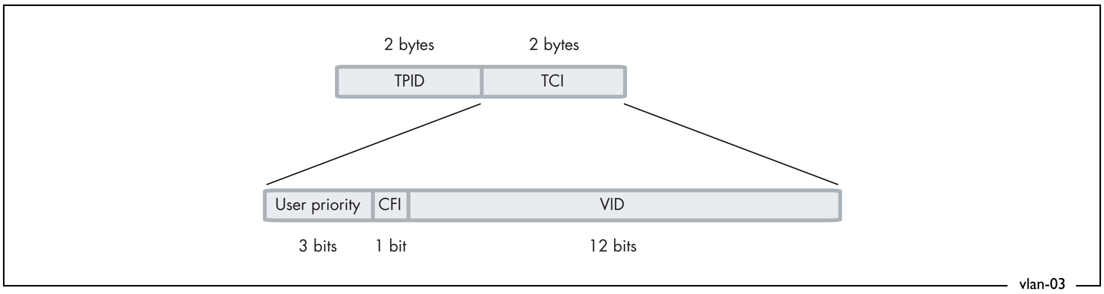
How is this achieved? How does one switch know that when it receives a broadcast packet that it associates to VLAN A that it must also forward that broadcast to other switches?

This can be done in a number of different ways, and in the early days of VLANs, just about every one of these ways was tried. Some vendors had their switches use a proprietary protocol to inform each other of their VLAN tables; some vendors used time-divided multiplexing in which different timeslots were allocated to different VLANs; other vendors used frame tagging.

In the end, frame tagging became the accepted standard. As we will see, in most respects this is a simple and elegant solution. However, it initially had one big downside: it required a fundamental change to format of the Ethernet header. This split the world's Ethernet devices into those that recognized tagged headers and those that did not recognize tagged headers. In other words, a lot of Ethernet equipment was rendered obsolete.

How does tagging work?

Simply, 4 bytes are inserted into the header of an Ethernet packet. This consists of 2 bytes of Tag Protocol Identifier (TPID) and 2 bytes of Tag Control Information (TCI):



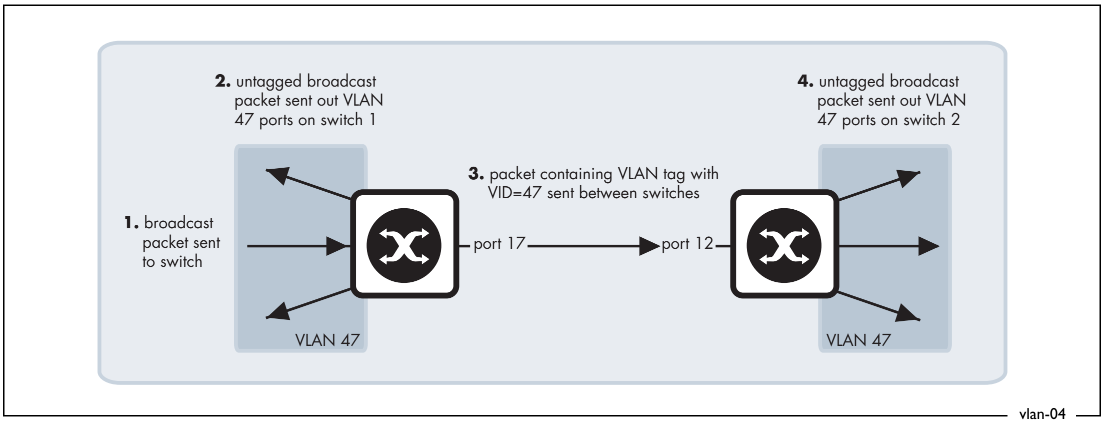
TPID is the tag protocol identifier, which indicates that a tag header is following and contains the user priority, canonical format indicator (CFI), and the VLAN ID.

User priority is a 3-bit field that allows priority information to be encoded in the frame. Eight levels of priority are allowed, where zero is the lowest priority and seven is the highest priority.

The CFI is a 1-bit indicator that is always set to zero for Ethernet switches. CFI is used for compatibility between Ethernet and Token Ring networks. If a frame received at an Ethernet port has a CFI set to 1, then that frame should not be bridged to an untagged port.

Then, the VID field contains the identifier of the VLAN. Actually, it is only the VID field that is really needed for distributing VLANs across switches—but the IEEE decided that they while they were altering the format of the Ethernet header, they might as well add the User Priority and CFI too.

Let us see how this tag makes it simple to distribute VLANs across switches. Consider a broadcast packet arriving at a switch port. By some criterion, the packet is associated with VLAN 47, i.e. a VLAN with VLAN ID=47. Now, port 17 of this switch is connected to port 12 of another switch that also has some ports in VLAN 47. The network administrator needs to configure port 17 of switch 1 and port 12 of switch 2 as "tagged" member ports of VLAN 47. This tells switch 1 to send the broadcast out port 17 as a tagged packet, with VID=47 in the tag. And it tells switch 2 to accept that tagged packet and associate it with VLAN 47. Then, switch 2 will send the packet out all its member ports of VLAN 47, because that is what it does with broadcasts that it has associated with VLAN 47.


2. untagged broadcast packet sent out VLAN 47 ports on switch 1
4. untagged broadcast packet sent out VLAN 47 ports on switch 2
3. packet containing VLAN tag with VID=47 sent between switches
1. broadcast packet sent to switch
port 17
port 12
VLAN 47
VLAN 47
vlan-04

The tag makes it very easy for the second switch to know what to do with the packet, because the tag marks this packet as belonging to VLAN 47, and switch 2 knows exactly what it should do with packets that belong to VLAN 47.

So, there really are only two simple rules:

- If a port is a tagged member of a VLAN, then any packets sent out that port by that VLAN must have a tag inserted into the header.
- If a tagged packet arrives in at a port, **and** the port is a tagged member of the VLAN corresponding to the VID in the packet's tag, then the packet is associated with that VLAN.

With these two simple rules, it is possible to distribute VLANs across multiple switches.

Mixing tagged and untagged packets on the same port

In the previous section, we discussed using tags to indicate the VLAN membership of packets that are transferred from one switch over to another. But, it is also possible that untagged packets will be transported across that link that joins the two switches.

For example, it could be that port 17 of switch 1 is an untagged member of VLAN 56, and port 12 of switch 2 is an untagged member of VLAN 56. In this case, if switch 1 needed to transport VLAN 56 packets over to switch 2, it would send them untagged. When those untagged packets arrived at switch 2, what VLAN would switch 2 decide to associate these packets with, given that they do not have a tag to indicate their VLAN membership? Well, in fact, switch 2 would realise that VLAN 56 is the untagged VLAN on the receiving port, so untagged packets would be deemed to belong to VLAN 56.

Obviously, a port can be an untagged member of only one port-based VLAN, otherwise there would be uncertainty about what VLAN incoming untagged packets belonged to. (Note, the situation is a bit different when subnet and protocol-based VLANs are introduced—see page 10 onwards). This VLAN is often referred to as the “native” VLAN of the port.

Often, you might not want to associate a native VLAN with the port that connects a switch to another switch, so that all packets coming into that port **must** use a VLAN tag to indicate their VLAN membership. This stops the switch from accepting any untagged packets on the port. In AlliedWare Plus, this is achieved by configuring a port to trunk mode and not configuring a native VLAN on it. In AlliedWare, it is achieved by setting the parameter **acceptable=vlan** on the port, so that the port will only accept VLAN-tagged packets.

Only accepting packets that match the port's VLAN configuration (ingress filtering)

Consider a port that is connected to a normal workstation. Normal applications on the workstation will never send tagged packets, so there is no requirement for the switch port to accept tagged packets.

But, is there any harm if the port does accept tagged packets if they happen to come along? Well, the answer is "quite possibly yes". If the workstation does send tagged packets, then it is very likely doing so for malicious reasons.

To guard against such maliciousness, most switches provide the ability to configure "ingress filtering". When ingress filtering is applied to a port, packets will only be accepted into a port if they match the VLAN configuration of that port. So, if the port is an untagged member of one VLAN, and nothing else, then only untagged packets will be accepted on the port. If the port is tagged for a set of VLANs, then a tagged packet will be accepted into the port only if it is tagged with the VID of one of the tagged VLANs configured on the port.

We highly recommend that you configure ingress filtering on all switch ports, because there is seldom a good reason for a port to accept packets from VLANs that are not configured on that port. Under AlliedWare Plus ingress filtering is enabled on all ports by default.

Other VLAN classification criteria

Up until now, we have been thinking just of port-based VLANs. However, there are other ways of defining VLAN membership. In this section, we will consider two examples of these other types of VLAN:

- Protocol-based VLANs
- Subnet-based VLANs

Protocol-based VLANs

With this method, different protocol types are assigned to different VLANs. For example, IP defines one VLAN, IPX defines another VLAN, Netbeui yet another VLAN, etc.

Protocol	VLAN
IP	1
IPX	2

Subnet-based VLANs

With this method, the VLAN membership is defined by the subnet to which a workstation's IP address belongs.

Subnet	VLAN
23.2.24.0	1
26.21.35.0	2

Workstation or packet?

Now that you have read the descriptions of protocol-based and subnet-based VLANs, it is possible that some awkward questions will come to your mind, like:

- Isn't a VLAN a set of workstations? How does a protocol specify a workstation? Surely a given workstation can send out packets using different protocols (often at the same time), depending on which applications it is running?

  and

- In the case of a subnet-based VLAN, which VLAN does a workstation belong to when it is not sending IP packets?

These are good questions.

At this point, you may be starting to see that the description of a VLAN as a set of workstations is a bit of a simplification. So, let us look a bit deeper here and get to a better understanding of what VLAN membership means.

In fact, a given workstation can belong to multiple VLANs. It could belong to one subnet-based VLAN when sending IP packets, another protocol-based VLAN when sending IPX packets, and yet another different port-based VLAN when sending some other protocol.

So, certainly, when analysing the VLAN setup on a network, it is a mistake to ask “what VLAN does this workstation belong to?” The more meaningful question to ask is “if a packet of such-and-such a protocol arrived at port x of the switch, which VLAN would that packet be associated with?”

It is important to really understand the change of mind-set that has just been introduced here. When initially learning about VLANs, it is usual to think of VLANs as sets of workstations. And, in practice, this is often all that a network administrator wants to achieve. However, once the VLAN configuration on a switch becomes complex, with multiple VLANs of different types all configured on the same port, it is no longer possible to really think about the VLAN from the workstation point of view. It becomes necessary to think of it from the packet point of view.

Therefore, it really is vital to think of packets being associated to VLANs when trying to understand VLAN configurations. Any other approach just ends in confusion.

The main point is that, when using protocol-based and subnet-based VLANs, it is data streams that are divided into VLANs, not necessarily whole workstations.

Some specific examples

Up until now, we have looked generically at how VLANs are implemented. In this section, we put this generic description into context by examining some specific examples.

Let us consider a case where multiple VLANs have been configured on a switch, and see what happens when certain packets arrive at the port.

VLAN setup on the switch

The switch uses the following VLANs:

- ports 1 - 4 of the switch are untagged members of the port-based VLAN 2.
- ports 3 - 6 of the switch are untagged members of the subnet-based VLAN 3, which is configured for the subnet 192.168.1.0/255.255.255.0.
- port 4 is an untagged member of the protocol-based VLAN 4, which is configured for protocols IP and IPX.
- port 5 is a tagged member of VLAN 2
- port 6 is a tagged member of VLAN 4

This switch implementation also has the following rules:

1. Subnet-based VLANs take precedence over protocol-based VLANs, which take precedence over port-based VLANs.
2. If a tagged packet arrives at a port, it is only accepted **if** that port is a tagged member of the VLAN corresponding to the VID in the packet's tag.

Treatment of packets

Now let us look at certain packets arriving at the switch:

A untagged IPX packet arrives at port 1

Port 1 is **only** a member of VLAN 2, so the packet will be associated with VLAN 2. The switch will look at the forwarding table for VLAN 2. If the destination MAC address of the packet is in the forwarding table, the packet will be forwarded out the corresponding port in that table entry. If the destination MAC address is not in the forwarding table for VLAN 2, then the packet will be flooded out **all** other ports of VLAN 2. So, it will be sent as an untagged packet out ports 2-4, and as a tagged packet out port 5.

An untagged IP packet with source/dest IP address in the 192.168.1.0/ 255.255.255.0 subnet arrives at port 4

Port 4 is a member of a subnet-based VLAN 3 configured for the subnet 192.168.1.0/ 255.255.255.0. So, the packet will be associated to VLAN 3. The switch will look at the forwarding table for VLAN 3. If the destination MAC address of the packet is in the forwarding table, the packet will be forwarded out the corresponding port in that table entry. If the destination MAC address is not in the forwarding table for VLAN 3, then the packet will be flooded out **all** other ports of VLAN 3. It will be sent as an untagged packet out ports 3, 5, and 6.

An untagged IP packet with source/dest IP address *not* in the 192.168.1.0/ 255.255.255.0 subnet arrives at port 4

Port 4 is a member of a subnet-based VLAN 3 configured for the subnet 192.168.1.0/ 255.255.255.0, but the packet's addresses are not in that subnet. So, the packet will not be associated with VLAN 3.

The next VLAN type in the precedence order is the protocol-based VLAN. Port 4 is a member of the protocol-based VLAN 4, configured for IP and IPX. As this is an IP packet, it will be associated with VLAN 4.

The switch only has one other port in VLAN 4. The packet will be sent as a tagged packet out port 6.

An untagged AppleTalk packet arrives at port 4

The AppleTalk packet cannot be associated with the subnet-based or the protocol-based VLANs on port 4, so it must drop through to the port-based VLAN on port 4.

So the packet is associated with VLAN 2. The switch will look at the forwarding table for VLAN 2. If the destination MAC address of the packet is in the forwarding table, the packet will be forwarded out the corresponding port in that table entry. If the destination MAC address is not in the forwarding table for VLAN 2, then the packet will be flooded out **all** other ports of VLAN 2. So, it will be sent as an untagged packet out ports 1-3, and as a tagged packet out port 5.

A tagged IPX packet arrives at port 4

Port 4 is an untagged member of the protocol-based VLAN 4, configured for IP and IPX. But, the packet is tagged, so it will be dropped.

A tagged packet with VID=10 arrives at port 5

Port 5 is a tagged member of VLAN 2. But the VID in the packet's tag does not match the VID of the VLAN (2), so the packet is dropped.

A tagged packet with VID=2 arrives at port 5

Port 5 is a tagged member of VLAN 2. The VID in the packet's tag matches the VID of the VLAN, so the packet is associated with VLAN 2.

The switch will look at the forwarding table for VLAN 2. If the destination MAC address of the packet is in the forwarding table, the packet will be forwarded out the corresponding port in that table entry. If the destination MAC address is not in the forwarding table for VLAN2, then the packet will be flooded out **all** other ports of VLAN 2. So, it will be sent as an untagged packet out ports 1-4.

USA Headquarters | 19800 North Creek Parkway | Suite 100 | Bothell | WA 98011 | USA | T: +1 800 424 4284 | F: +1 425 481 3895
European Headquarters | Via Motta 24 | 6830 Chiasso | Switzerland | T: +41 91 69769.00 | F: +41 91 69769.11
Asia-Pacific Headquarters | 11 Tai Seng Link | Singapore | 534182 | T: +65 6383 3832 | F: +65 6383 3830
www.alliedtelesis.com