VLAN

What is a VLAN?

VLAN is a virtual LAN. In technical terms, a VLAN is a broadcast domain created by switches. Normally, it is a router creating that broadcast domain. With VLAN's, a switch can create the broadcast domain.
This works by, you, the administrator, putting some switch ports in a VLAN other than 1, the default VLAN. All ports in a single VLAN are in a single broadcast domain.
Because switches can talk to each other, some ports on switch A can be in VLAN 10 and other ports on switch B can be in VLAN 10. Broadcasts between these devices will not be seen on any other port in any other VLAN, other than 10. However, these devices can all communicate because they are on the same VLAN. Without additional configuration, they would not be able to communicate with any other devices, not in their VLAN.

Are VLANs required?

It is important to point out that you don't have to configure a VLAN until your network gets so large and has so much traffic that you need one. Many times, people are simply using VLAN's because the network they are working on was already using them.
Another important fact is that, on a Cisco switch, VLAN's are enabled by default and ALL devices are already in a VLAN. The VLAN that all devices are already in is VLAN 1. So, by default, you can just use all the ports on a switch and all devices will be able to talk to one another.

When do I need a VLAN?

You need to consider using VLAN's in any of the following situations:

- You have more than 200 devices on your LAN
- You have a lot of broadcast traffic on your LAN
- Groups of users need more security or are being slowed down by too many broadcasts?
- Groups of users need to be on the same broadcast domain because they are running the same applications. An example would be a company that has VoIP phones. The users using the phone could be on a different VLAN, not with the regular users.
- Or, just to make a single switch into multiple virtual switches.

Why not just subnet my network?

A common question is why not just subnet the network instead of using VLAN's? Each VLAN should be in its own subnet. The benefit that a VLAN provides over a subnetted network is that devices in different physical locations, not going back to the same router, can be on the same network. The limitation of subnetting a network with a router is that all devices on that subnet must be connected to the same switch and that switch must be connected to a port on the router.

What is a trunk port?

A trunk connection is a link that carries VLAN information between *VLAN-aware* Layer 2 devices.

These devices could be two switches, a switch and a router, or even a switch and an end station. The advantage of the trunk is that through one connection, many VLANs can be transported between the two switches; therefore we do not have to implement a dedicated (and costly) connection for each VLAN. Trunking can dramatically improve the performance, manageability, and reliability for applications.

For example, let us assume, we have connected a link between the ports of two switches. If the switch ports defined on the switches are members of the same VLAN, the ports will pass any traffic only for the VLAN associated with their port connections. By default, the ports are in a non- trunk mode called an *Access* link. If you want the traffic to pass between multiple VLANs established on multiple switches, you will need to first establish a trunk connection between the switch ports.
A trunk port must run a special trunking protocol. The protocol used would be Cisco's proprietary Inter-switch link (ISL) or the IEEE standard 802.1q.

VLAN configuration With Router

192.168.1.100	PC-1	VLAN 10	192.168.1.1
192.168.2.100	PC-2	VLAN 20	192.168.2.1

Switch Configuration:

```
enable
configuration terminal
vlan 10
vlan 20
show vlan
config t
int fa 0/1
switchport mode access
switchport access vlan 10
int fa 0/2
switchport mode access
switchport access vlan 20
ctrl Z
show vlan
copy run start
```

Setting up Trunk

```
config t
int fa 0/24
switchport mode trunk
end
copy run start
```

Router Configuration

```
enable
config t
int fa 0/0.1
encapsulation dot1q 10
ip address 192.168.1.1          255.255.255.0
no shut
int fa 0/0.2
encapsulation dot1q 20
ip address 192.168.2.1          255.255.255.0
no shut
show run
show ip route
```