Wireless and Beyond

ARP

- The *Address Resolution Protocol*
- Usually considered to be a part of the link layer (shows the layering model has some problems)
- Part of the Internet Protocol (IP) to solve a specific problem
- The physical layer has (e.g., 6 byte Ethernet) addresses, while the network layer has independent (4 byte) IP addresses

Wireless and Beyond

ARP

- What is the connection between them?
- There is no a priori connection as they live in different layers
- But they need some connection otherwise we won't be able to use use a physical layer to send IP packets: we won't know where a particular IP packet should physically be sent

Wireless and Beyond

ARP

- Every host on the Internet has (at least) one 32 bit (4 byte) IP address
- It is unique to that host and so identifies that machine on the Internet
- (We are ignoring some considerations to be come clearer later)

Wireless and Beyond

ARP

- Conventionally the addresses are written as a *dotted quad* of decimal values
- Thus

00000001000000100000001100000100

is written

1.2.3.4

Wireless and Beyond

ARP

- IP addresses are chosen by the local system administrator to suit the local network
- Ethernet addresses are built into the interface hardware by the manufacturer
- The two addresses bear absolutely no relationship to one another (as we would expect from the layering principles)

Wireless and Beyond

ARP

- Suppose want to send a packet over (say) an Ethernet
- We only know the destination's IP address
- To build the Ethernet frame we have to know the Ethernet address that the destination has
- This is what ARP does: find the hardware address corresponding to an IP address

Wireless and Beyond

ARP

- ARP broadcasts an *ARP Request* packet that contains the target IP address in an Ethernet frame with destination address ff:ff:ff:ff:ff:ff (and source its own Ethernet address)
- All hosts on the local network read the frame
- The target host recognises the request for its IP address

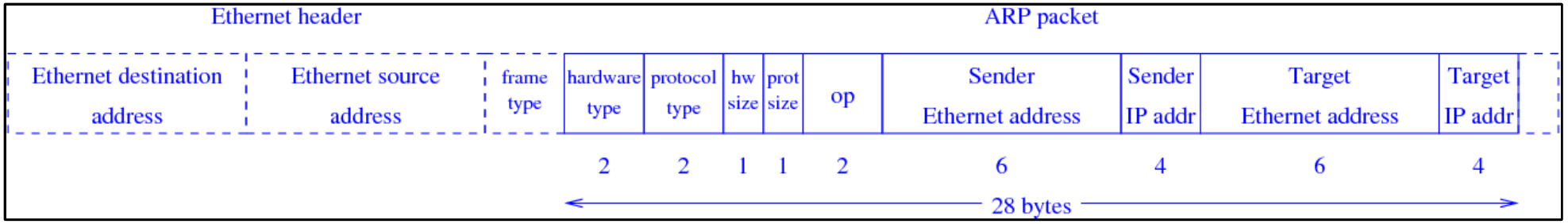
Wireless and Beyond

ARP

- The target sends an *ARP Reply* packet containing its own Ethernet address (the other hosts need do nothing)
- It knows the source's Ethernet address as read from the request packet
- The source gets the reply and reads out the target's Ethernet address
- It can now use that Ethernet address to send IP packets

Wireless and Beyond

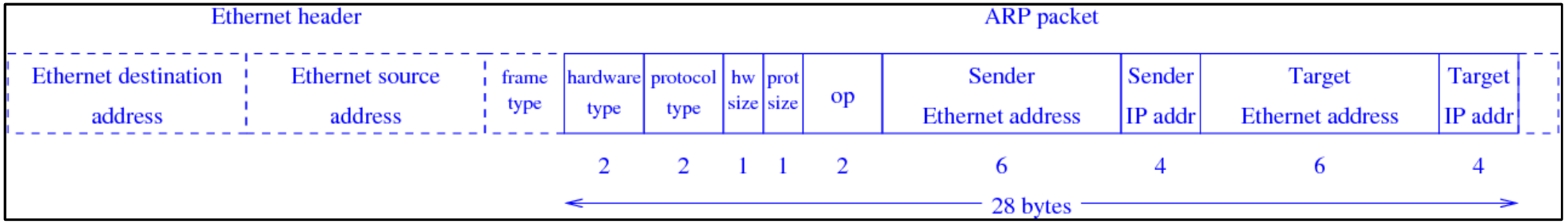
ARP



- The frame type for ARP is 0806
- Hardware type: 1 for an Ethernet address
- Protocol type: 0800 for an IP address
- Sizes: sizes in bytes of the address fields, 6 for Ethernet, 4 for IP

Wireless and Beyond

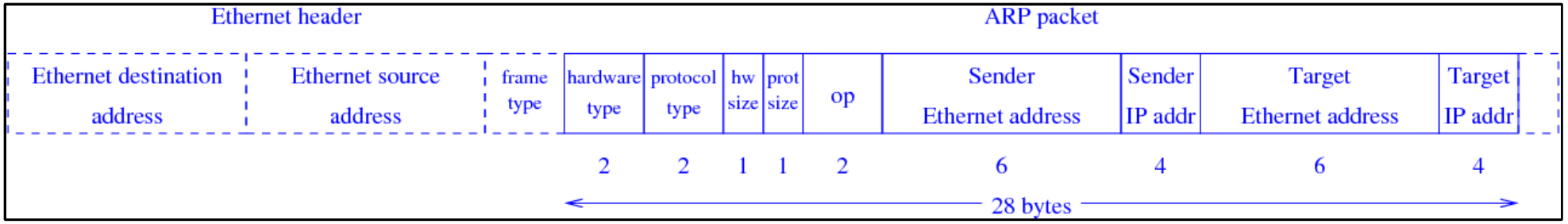
ARP



- OP: 1 for a request, 2 for a reply
- Address fields: the data
- In a request the destination hardware field is not filled in as this is what we are trying to find!

Wireless and Beyond

ARP



- In a reply the sender Ethernet address is the address we seek

Wireless and Beyond

ARP

- The source caches the address so it doesn't need to do an ARP for *every* IP packet
- The cache expires after (say) 20 minutes, after which a fresh ARP exchange is needed
- This is in case the Ethernet-to-IP address relationship changes, e.g., an IP address is reassigned to a new machine

Wireless and Beyond

ARP

- If no machine on the local network has the requested IP address, or that machine is down, no reply will be forthcoming
- In this case, after a few seconds (and a few repeated ARP requests), an error message is send to the application trying to make the IP connection
- This might be “no such host” or “host unreachable”

Wireless and Beyond

Gratuitous ARP

- It is sometimes useful to give an ARP reply even if nobody has asked for it, for example a new machine joins the network or an existing machine changes its IP address for some reason
- This is a *gratuitous ARP*
- All machines on the local network are free to read any ARP reply and modify their own ARP caches accordingly

Wireless and Beyond

Gratuitous ARP

- So a gratuitous ARP would help break old associations that are no longer valid but still cached
- Without a gratuitous ARP a host might send an IP packet to the old, wrong address

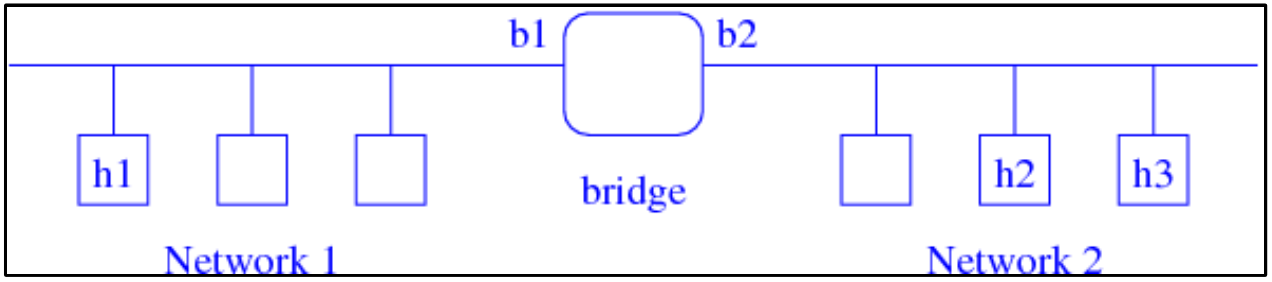
Wireless and Beyond

ARP Bridging

- A clever trick that shows ARP can be used for things other than it was designed to do
- This trick allows us to extend an Ethernet (or other network) over a physically larger distance than its specifications allow, and to join a wireless network to a wired one so they appear to be a single network

Wireless and Beyond

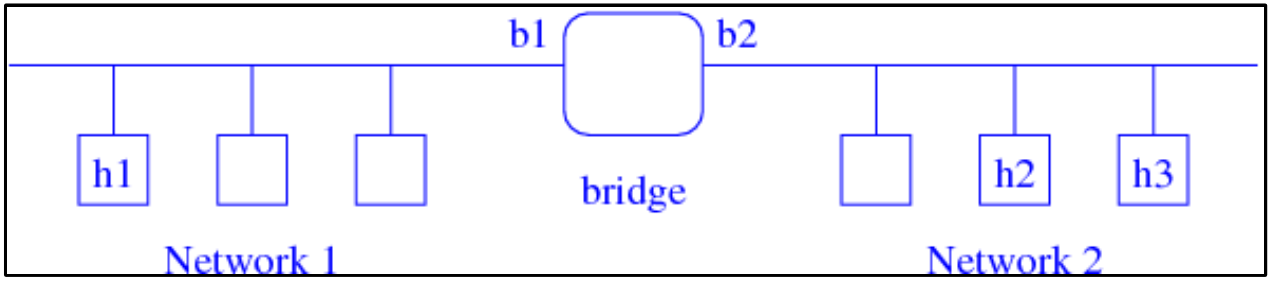
ARP Bridging



- A *bridge* is a host with two interfaces, one on each network
- If host h1 wishes to send to host h2 it must determine its hardware address

Wireless and Beyond

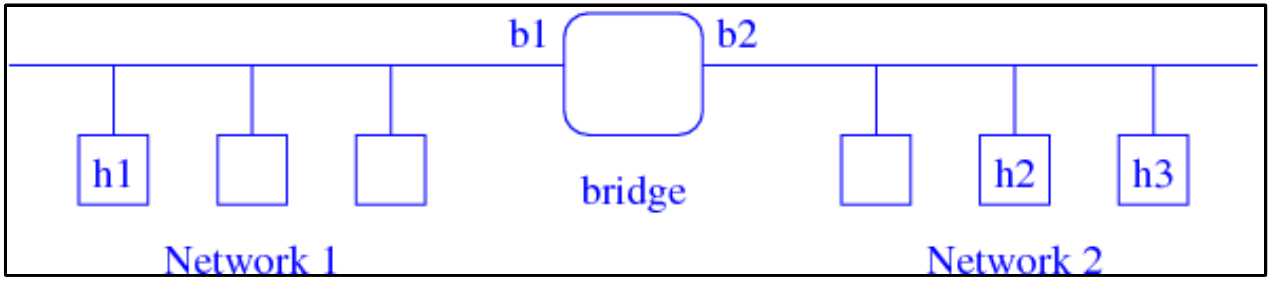
ARP Bridging



- So h1 sends an ARP broadcast for h2
- The bridge sees this request and responds on behalf of h2 (a *proxy* ARP), but it supplies its *own* hardware address b1

Wireless and Beyond

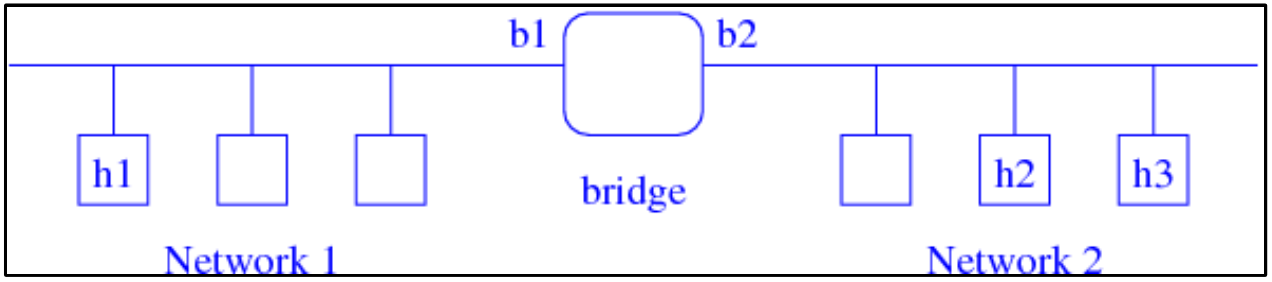
ARP Bridging



- Now h1 sends data to what it thinks is h2, but is actually the bridge
- The bridge reads the packet, sees it is destined for h2 (by its IP address) and forwards it to the other network where h2 can read it

Wireless and Beyond

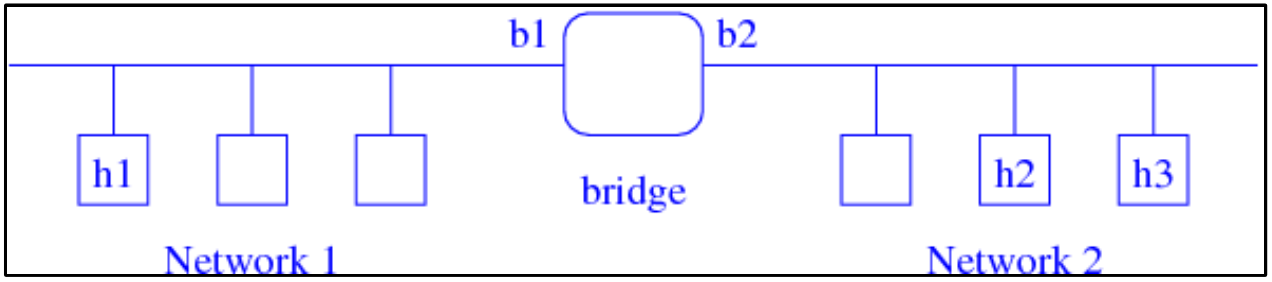
ARP Bridging



- The forwarded frame will be changed to have h2 as destination and b2 as source
- If h2 replies, it can either use b2 which it got from the original packet or do an ARP request, which the bridge proxies in a symmetrical way

Wireless and Beyond

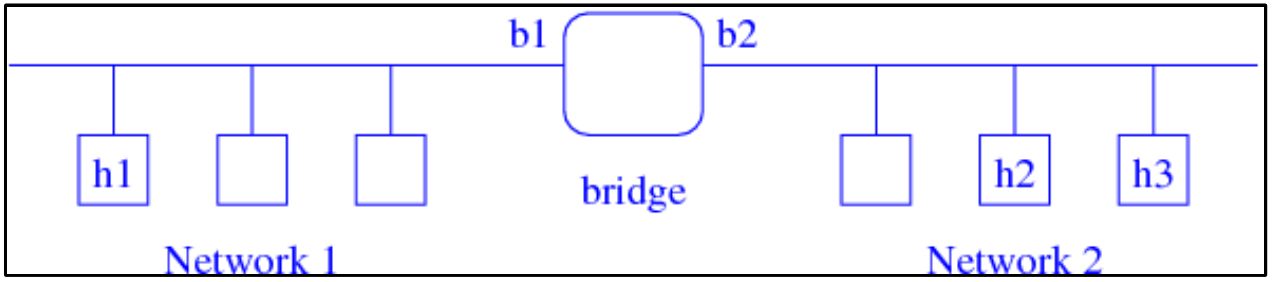
ARP Bridging



- In either case the packet goes to the bridge, which forwards it to h1, again rewriting the frame addresses appropriately
- This is all transparent to h1 and h2 who believe they are on the same network

Wireless and Beyond

ARP Bridging



- This is sometimes called *transparent* bridging
- If h1 is communicating with both h2 and h3 its cache will show then to have the *same* hardware address b1: this is not a problem

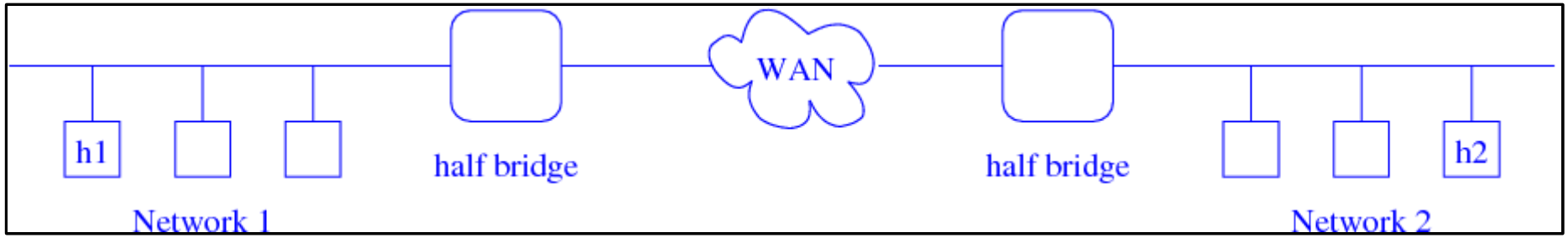
Wireless and Beyond

ARP Bridging

- ARP bridging is fine for joining a pair of small networks, but less so for larger collections of networks
- IEEE 802.1d Ethernet Bridging standard addresses this, dealing with the cases of multiple routes between hosts

Wireless and Beyond

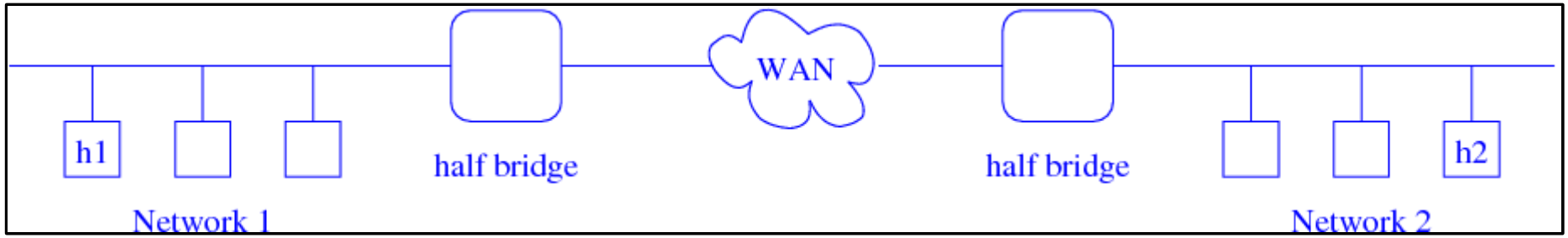
Remote Bridging



- Bridging can also connect networks that are widely separated, e.g., by a WAN, again to appear as a single network using *remote bridging*
- Compare this with *tunnelling*

Wireless and Beyond

Remote Bridging



- The endpoints are called *half bridges*
- This is similar in principle to local bridging, but now the half bridges must cope with encapsulation over the WAN; differences in speed and packet sizes of the LAN and WAN and so on

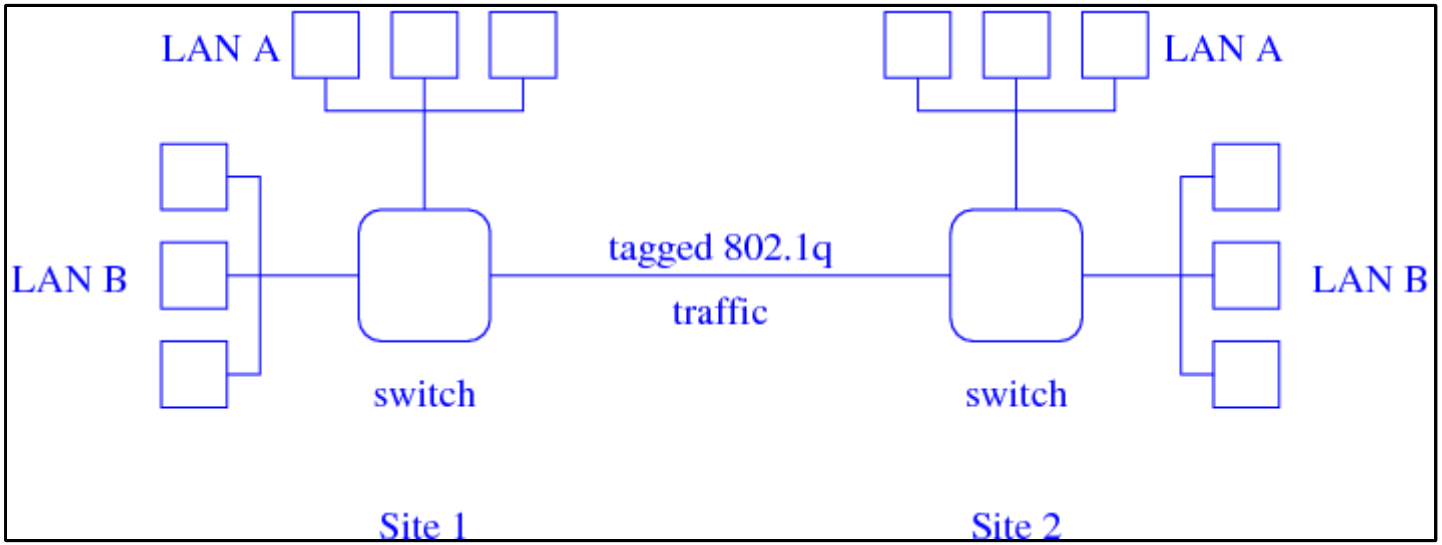
Wireless and Beyond

Virtual Bridging

- Another variation is 802.1q *virtual bridging*
- This allows more than one network to run on a single physical network

Wireless and Beyond

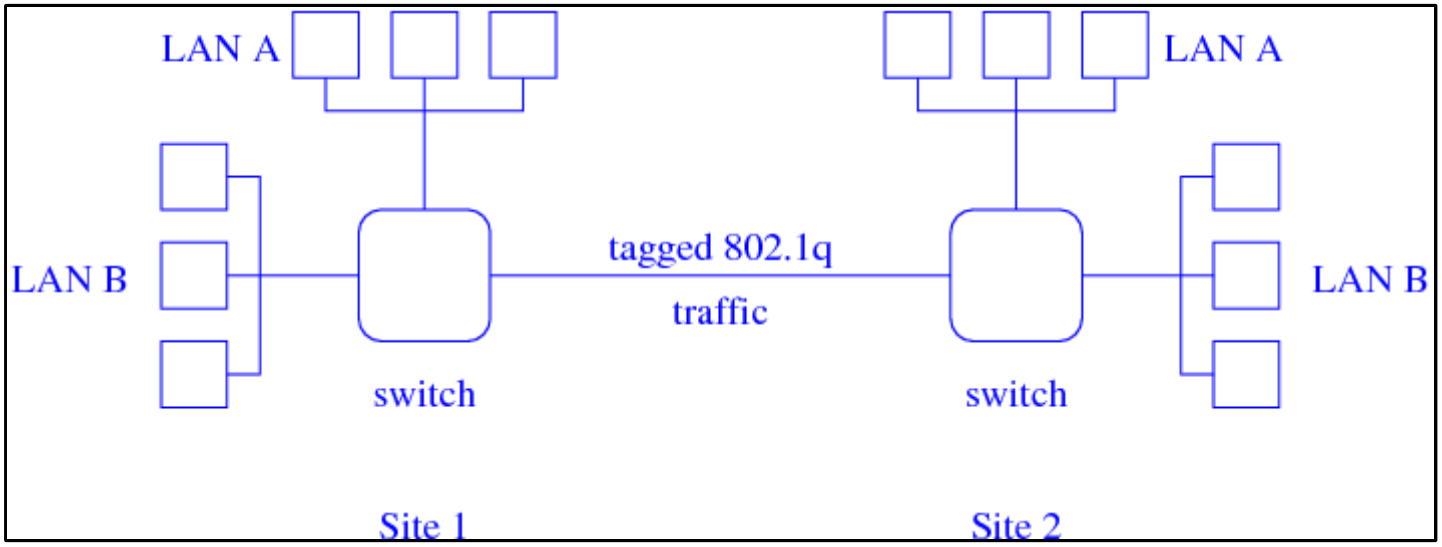
Virtual Bridging



- A company has two separate sites 1 and 2 with a single dedicated link between them

Wireless and Beyond

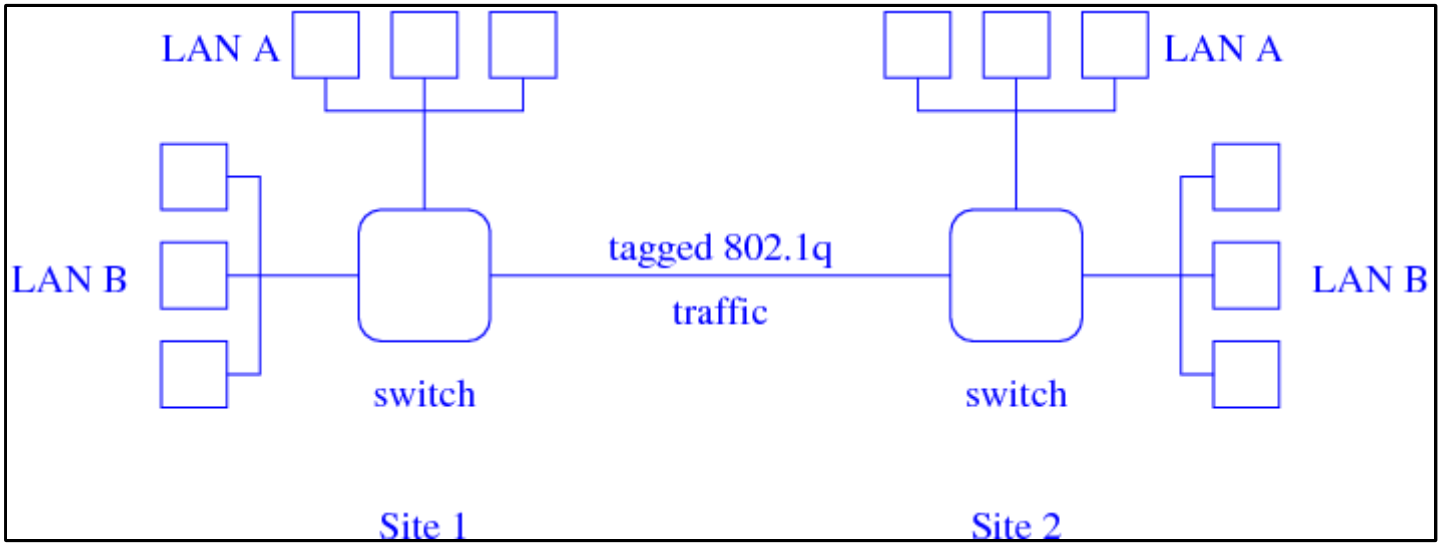
Virtual Bridging



- They want to run two separate LANs, A and B but not to buy a second link between the sites

Wireless and Beyond

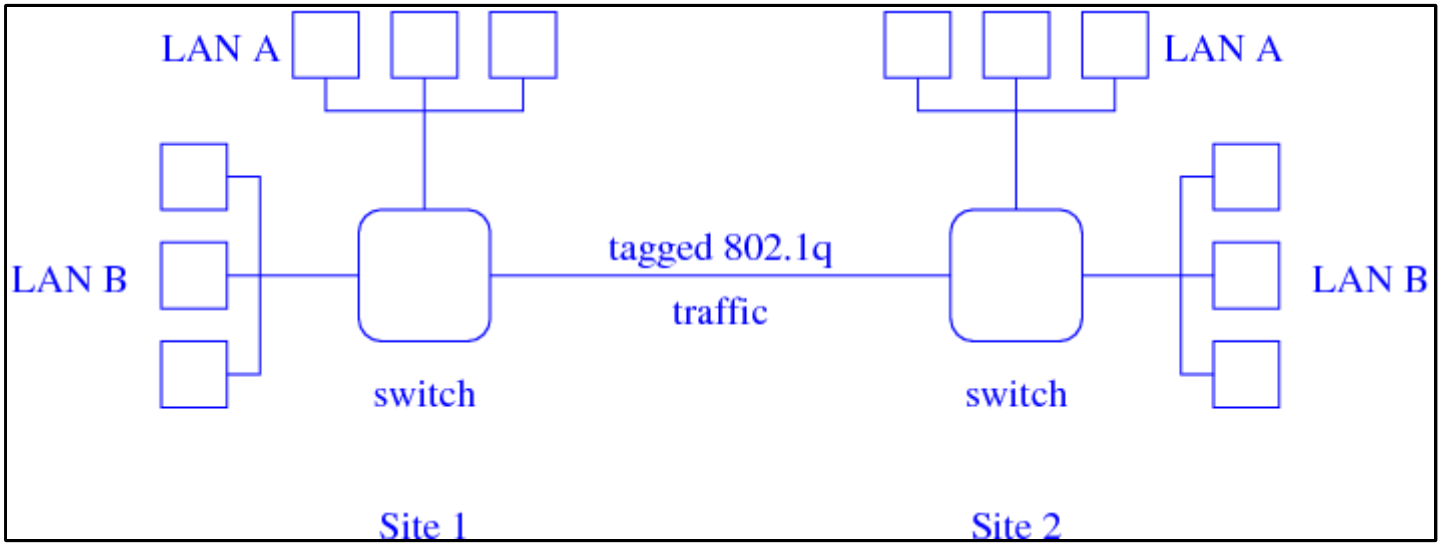
Virtual Bridging



- They can use 802.1q *tagging*
- A packet from LAN A, say, arrives at the switch

Wireless and Beyond

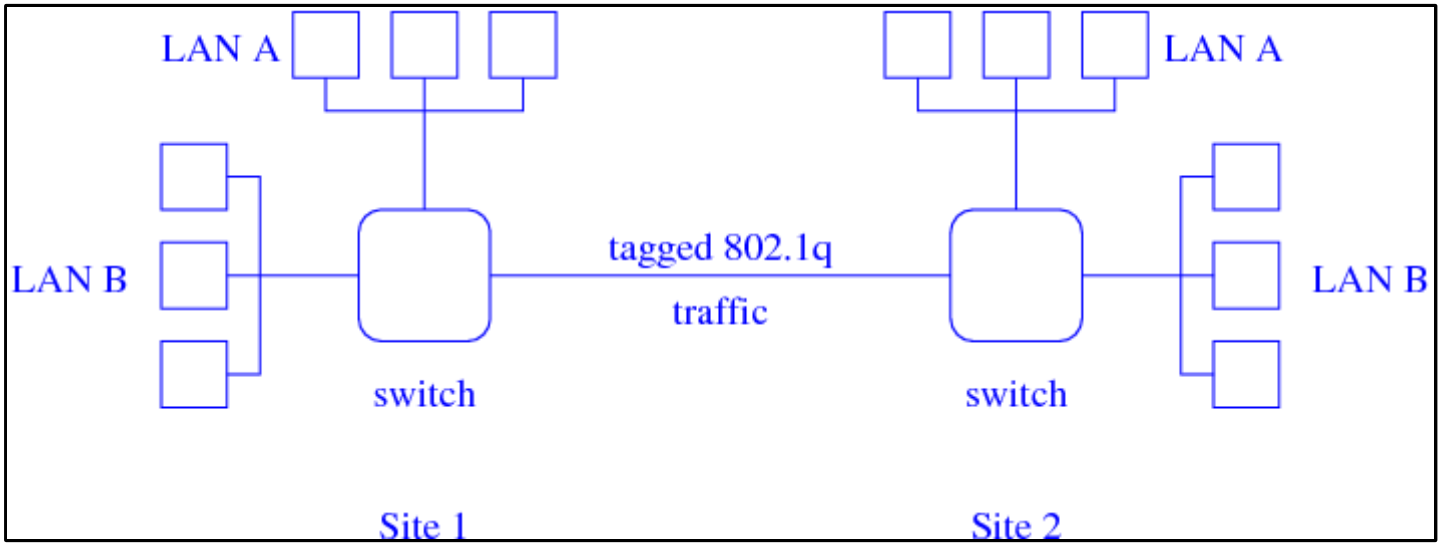
Virtual Bridging



- The switch knows to route the packet over the remote link: it places a 802.1q *tag* on the frame

Wireless and Beyond

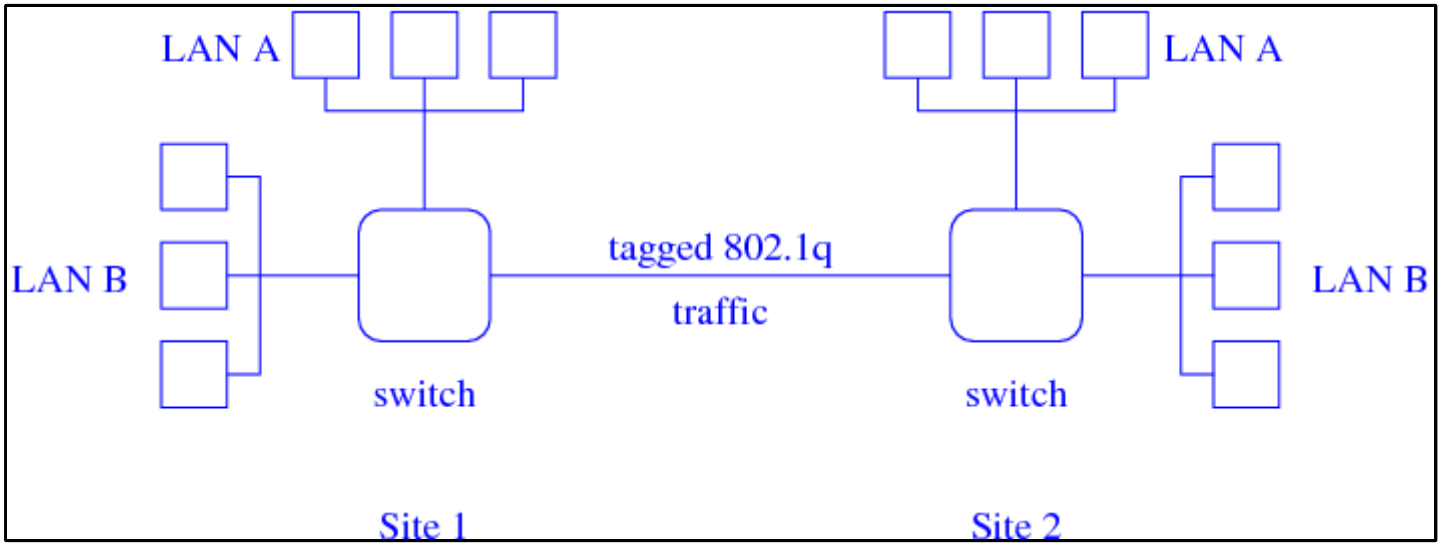
Virtual Bridging



- A tag is an extra four byte header containing a *Virtual LAN Identifier* (VID), a 12 bit integer

Wireless and Beyond

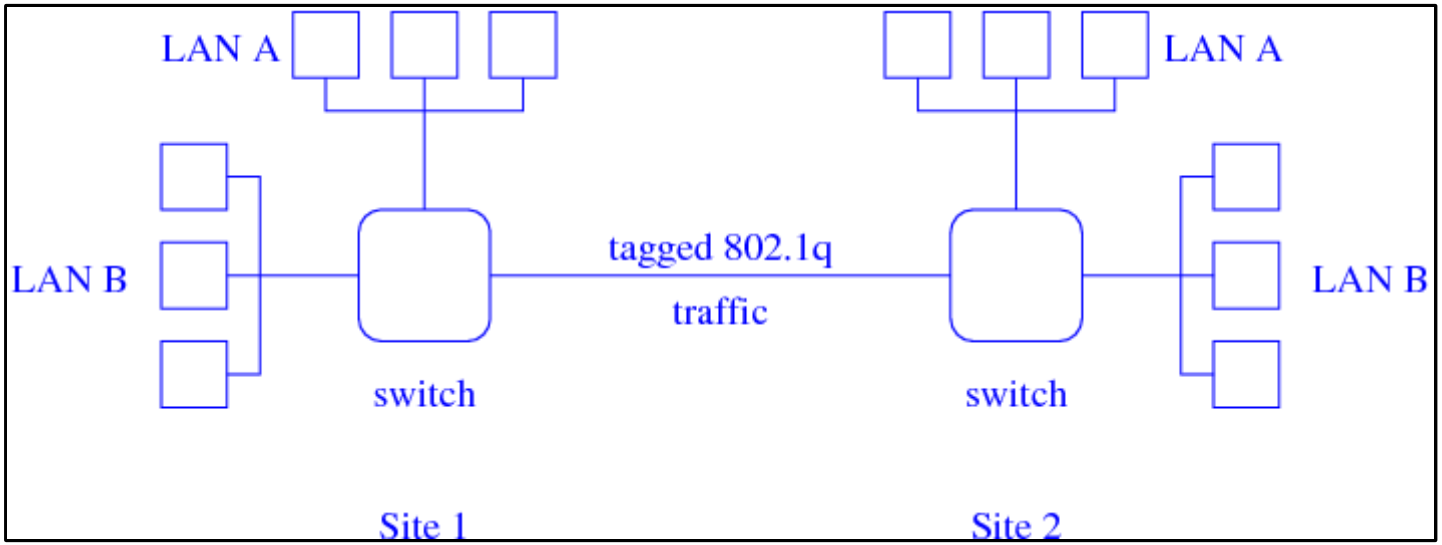
Virtual Bridging



- The frame type is changed from 0800 to 8100 to indicate a tagged packet

Wireless and Beyond

Virtual Bridging



- The switch in site 2 receives the packet, sees the tag, removes it and forwards the packet to LAN A

Wireless and Beyond

Virtual Bridging

- This generalises well to many virtual LANs (VLAN) and allows many networks to share infrastructure, thus saving on cost

Wireless and Beyond

Bridging

- Bridging is useful, but shouldn't be taken too far
- Larger networks have more and more traffic
- Just think of the ARP broadcasts alone!
- It is better to split a large network into several smaller ones: see subnetting, later

Wireless and Beyond

RARP

- *Reverse ARP* addresses to opposite problem to ARP: given a hardware address find the IP address
- Needed by hosts that don't initially know their own IP address, e.g., a diskless computer, a laptop plugging into a network, a refrigerator, etc.

Wireless and Beyond

RARP

- Very similar to ARP
- Frame type 8035
- Same frame layout
- Op type 3 for a RARP request, type 4 for a RARP reply
- RARP is OK for limited purposes, but much better protocols exist to solve the same problem (see DHCP)