Networks

IP Routing

A quick note regarding when the destination is not on the local network

Networks

IP Routing

A quick note regarding when the destination is not on the local network

IP routing for the source host is quite simple: if the destination is on the local network, send the packet directly. This probably uses ARP (on the first packet) to get the hardware address of the destination

Networks

IP Routing

If the destination is *not* on the local network, to solution is to send the packet to a *gateway* host and let it deal with where to send it next

Networks

IP Routing

If the destination is *not* on the local network, to solution is to send the packet to a *gateway* host and let it deal with where to send it next

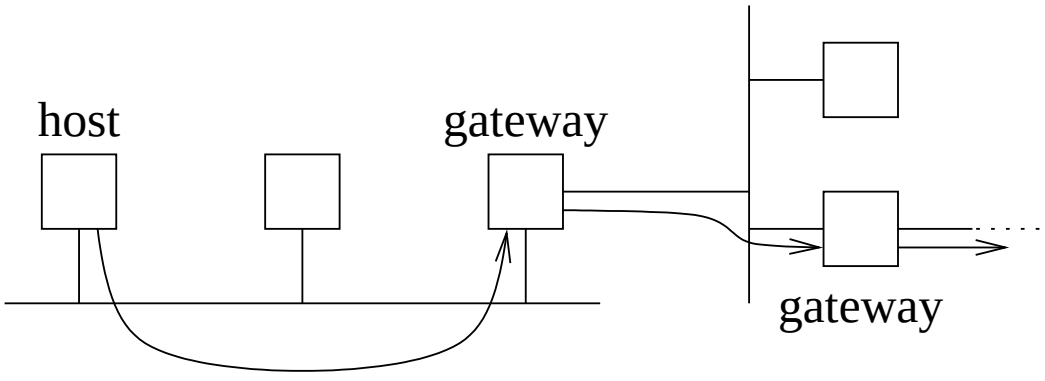
A gateway is just a machine on more than one network

Networks

IP Routing

If the destination is *not* on the local network, to solution is to send the packet to a *gateway* host and let it deal with where to send it next

A gateway is just a machine on more than one network



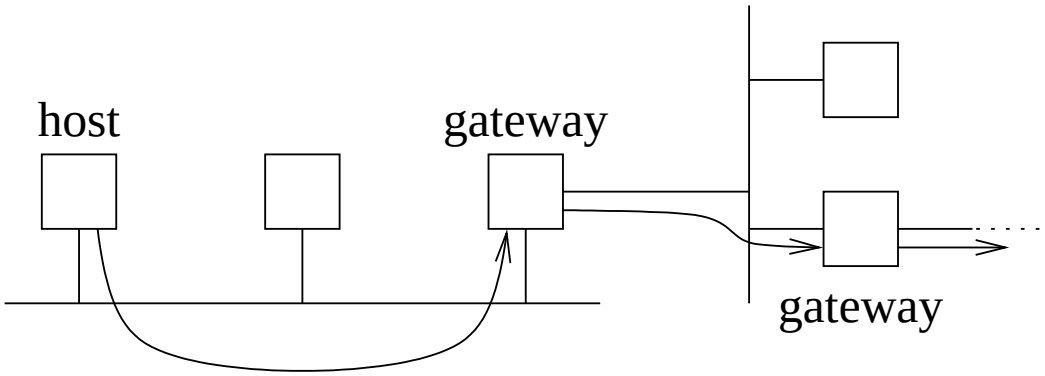
Gateway

Networks

IP Routing

If the destination is *not* on the local network, to solution is to send the packet to a *gateway* host and let it deal with where to send it next

A gateway is just a machine on more than one network



Gateway

This keeps the complexity of the software needed on the hosts down: only the gateway will need to have a bit of intelligence about routing

Networks

IP Routing

So information a source host needs to know includes:

- its own address and network
- the address of a gateway machine

Networks

IP Routing

So information a source host needs to know includes:

- its own address and network
- the address of a gateway machine

We shall see later how it gets this information

Networks

IP Routing

So, for a host the routing software is:

Networks

IP Routing

So, for a host the routing software is:

- is the destination on the local network?

Networks

IP Routing

So, for a host the routing software is:

- is the destination on the local network?
- yes: send it directly, possibly with an ARP, if needed

Networks

IP Routing

So, for a host the routing software is:

- is the destination on the local network?
- yes: send it directly, possibly with an ARP, if needed
- no: send it to the gateway, possibly with an ARP, if needed

Networks

IP Routing

So, for a host the routing software is:

- is the destination on the local network?
- yes: send it directly, possibly with an ARP, if needed
- no: send it to the gateway, possibly with an ARP, if needed

Note in the latter case, the host might need to do an ARP for the *gateway*

Networks

IP Routing

In the non-local case, the packet is going to the gateway, so we would need to ARP for the hardware address of the *gateway*

Networks

IP Routing

In the non-local case, the packet is going to the gateway, so we would need to ARP for the hardware address of the *gateway*

The packet, with IP address of the final destination, is put into a frame with Ethernet address of the gateway

Networks

IP Routing

In the non-local case, the packet is going to the gateway, so we would need to ARP for the hardware address of the *gateway*

The packet, with IP address of the final destination, is put into a frame with Ethernet address of the gateway

Since the packet needs to go to the gateway

Networks

IP Routing

In the non-local case, the packet is going to the gateway, so we would need to ARP for the hardware address of the *gateway*

The packet, with IP address of the final destination, is put into a frame with Ethernet address of the gateway

Since the packet needs to go to the gateway

So, here, the physical and network addresses in the Ethernet frame are completely unrelated!

Networks

IP Routing

This is another reason why we need both hardware and software addresses

Networks

IP Routing

This is another reason why we need both hardware and software addresses

The IP address is for the ultimate destination; the hardware address is for the next hop

Networks

IP Routing

ARP is not restricted to Ethernet and IP, but can be used to pair any physical and network layer addresses

Networks

IP Routing

ARP is not restricted to Ethernet and IP, but can be used to pair any physical and network layer addresses

Exercise Is ARP needed on a PPP connection?

ARP

ARP is a simple protocol

ARP

ARP is a simple protocol

On an Ethernet, the ARP broadcast has to be put in an Ethernet frame, so what destination address does it put on the frame?

ARP

ARP is a simple protocol

On an Ethernet, the ARP broadcast has to be put in an Ethernet frame, so what destination address does it put on the frame?

It broadcasts an *ARP Request* packet (protocol number 0806) in an Ethernet frame with destination hardware address `ff:ff:ff:ff:ff:ff` and source its own Ethernet address

ARP

ARP is a simple protocol

On an Ethernet, the ARP broadcast has to be put in an Ethernet frame, so what destination address does it put on the frame?

It broadcasts an *ARP Request* packet (protocol number 0806) in an Ethernet frame with destination hardware address `ff:ff:ff:ff:ff:ff` and source its own Ethernet address

All hosts on the local network read the frame

ARP

ARP is a simple protocol

On an Ethernet, the ARP broadcast has to be put in an Ethernet frame, so what destination address does it put on the frame?

It broadcasts an *ARP Request* packet (protocol number 0806) in an Ethernet frame with destination hardware 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

ARP

The target sends an *ARP Reply* packet (in a normal Ethernet frame) containing its own Ethernet address

ARP

The target sends an *ARP Reply* packet (in a normal Ethernet frame) containing its own Ethernet address

It knows the source's Ethernet address as read from the request packet

ARP

The target sends an *ARP Reply* packet (in a normal Ethernet frame) containing its own Ethernet address

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

ARP

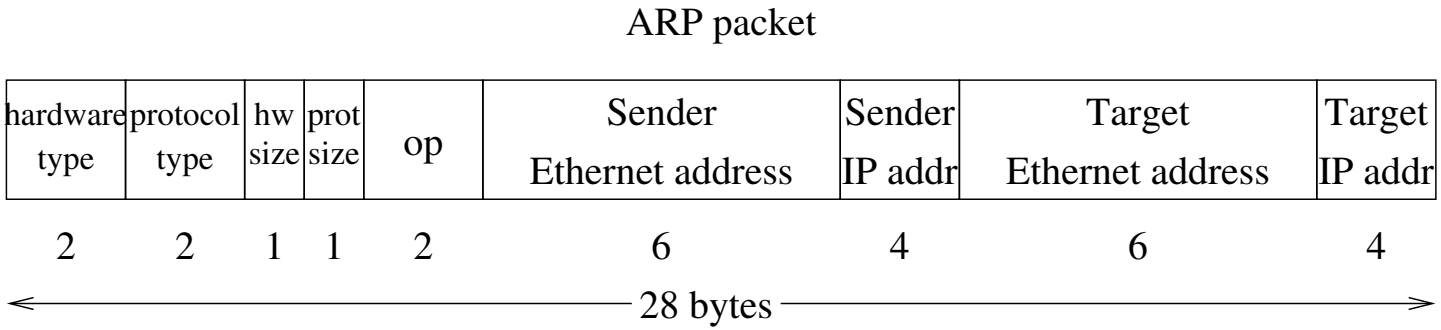
The target sends an *ARP Reply* packet (in a normal Ethernet frame) containing its own Ethernet address

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

The other hosts on the network need do nothing

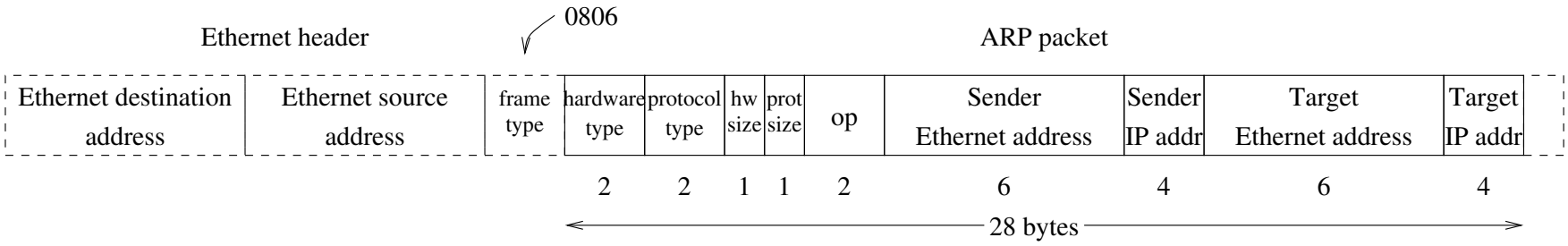
ARP



ARP packet

The Ethernet frame type for ARP is 0806

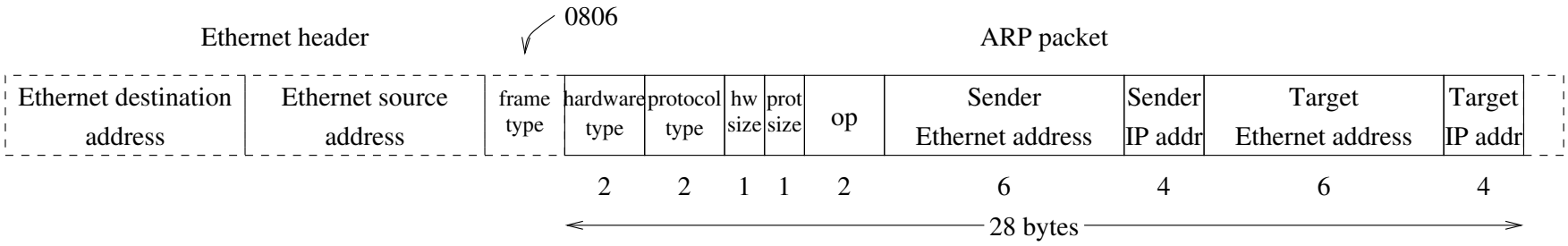
ARP



ARP packet within Ethernet frame

Contained within an Ethernet frame

ARP

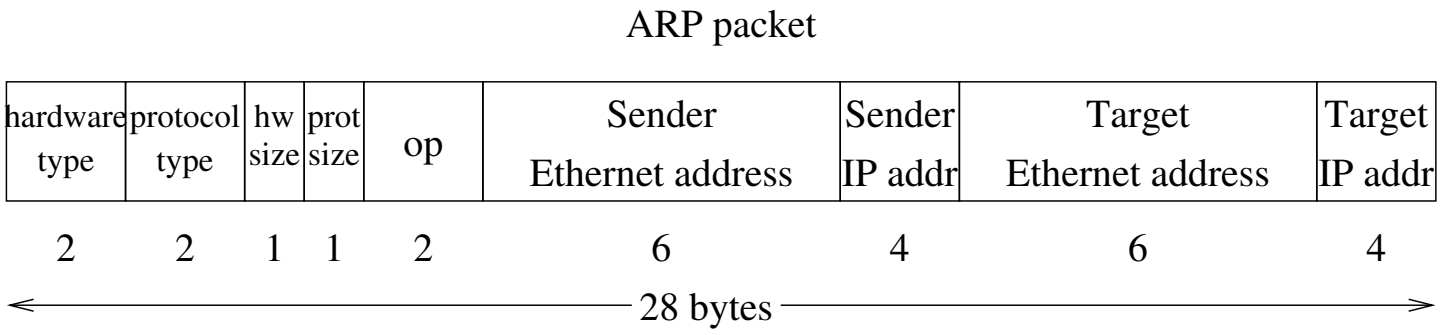


ARP packet within Ethernet frame

Contained within an Ethernet frame

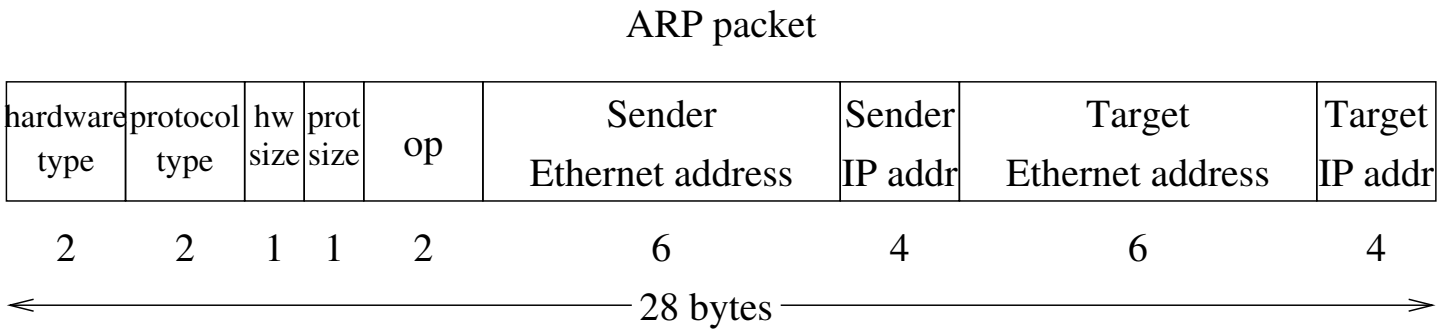
The Ethernet type field allows the software that reads the packet from the Ethernet card to pass the contents of the packet to the software that implements ARP

ARP



ARP fields

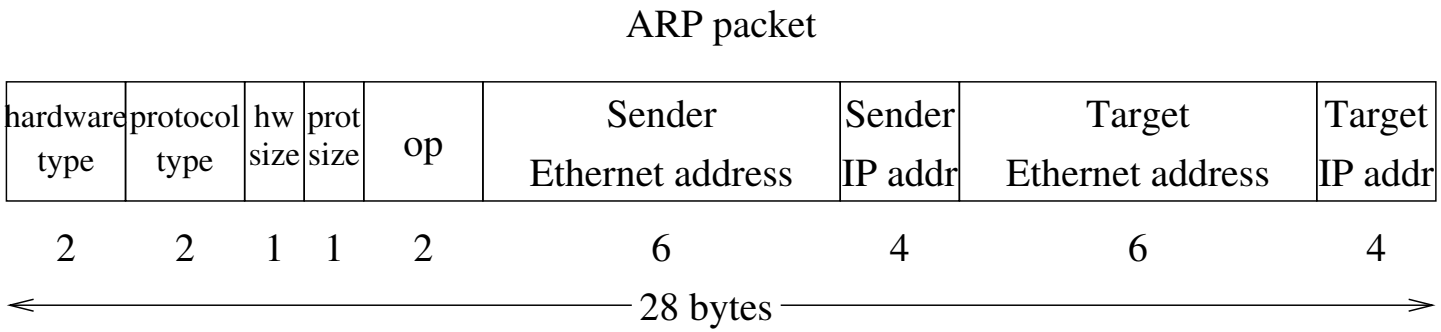
ARP



ARP fields

- Hardware type: 1 for an Ethernet address

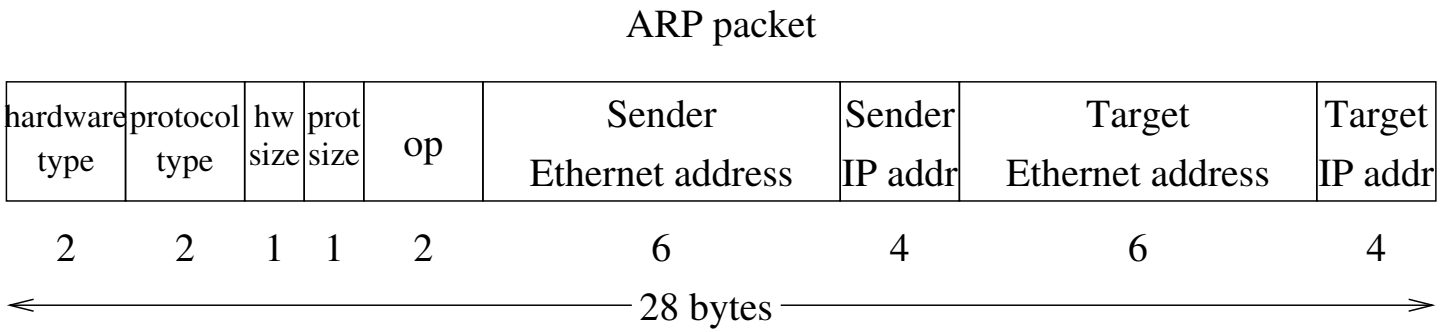
ARP



ARP fields

- Hardware type: 1 for an Ethernet address
- Protocol type: 0800 for an IP (version 4) address

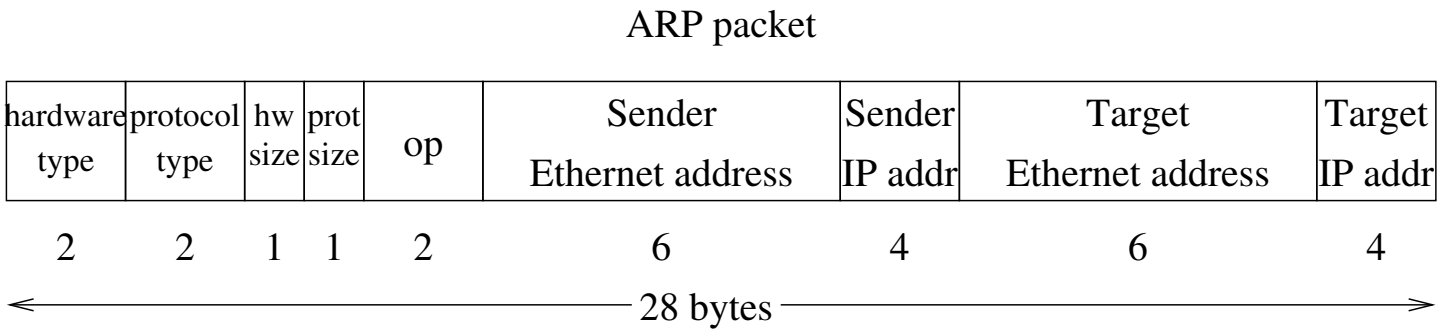
ARP



ARP fields

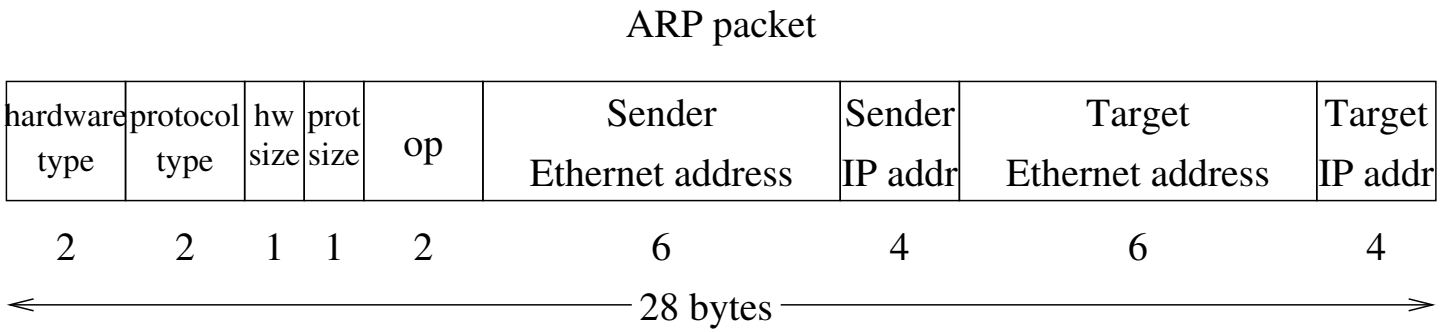
- Hardware type: 1 for an Ethernet address
- Protocol type: 0800 for an IP (version 4) address
- Sizes: sizes in bytes of the address fields, 6 for Ethernet, 4 for IP

ARP



ARP fields

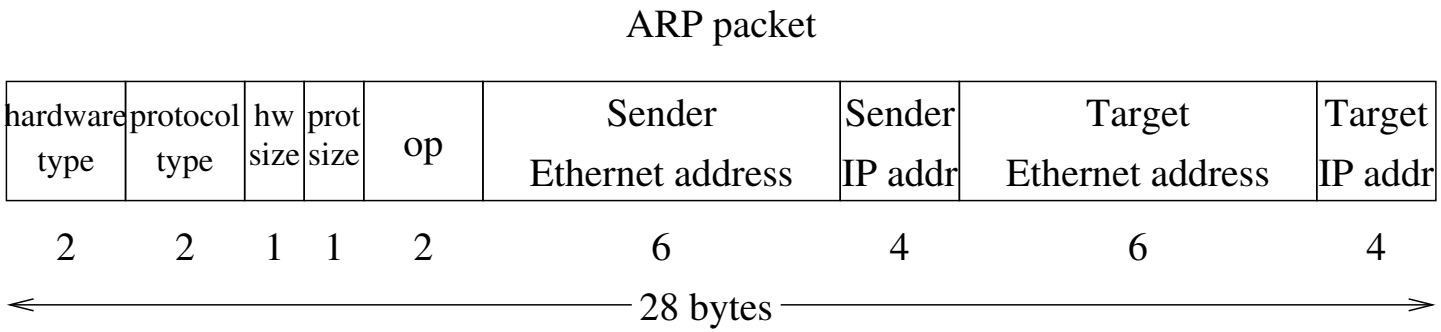
ARP



ARP fields

- OP: 1 for a request, 2 for a reply

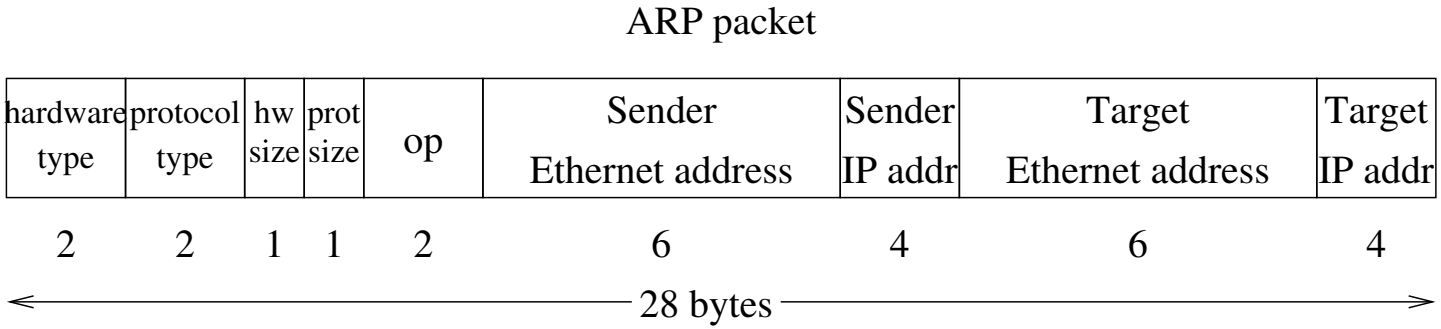
ARP



ARP fields

- OP: 1 for a request, 2 for a reply
- Address fields, with lengths as given: the data

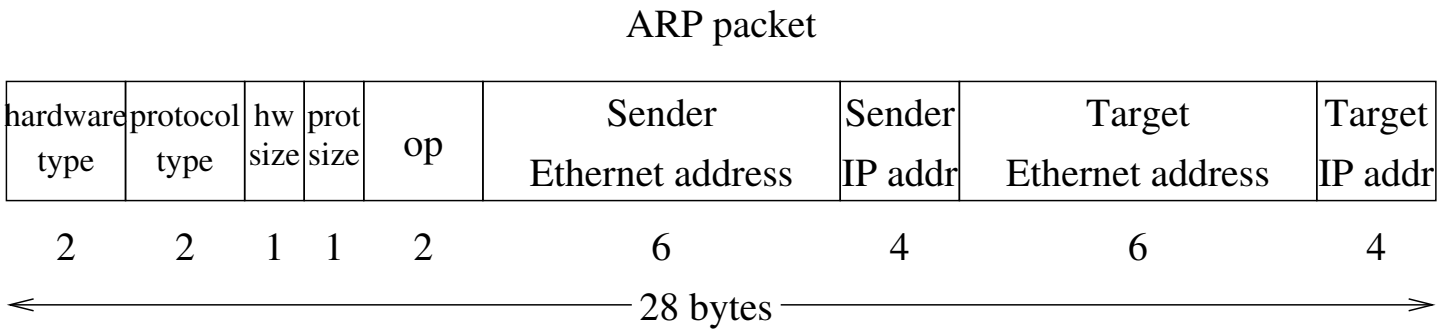
ARP



ARP fields

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

ARP



ARP fields

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

ARP

If no machine on the local network has the requested IP address, or that machine is down, no reply will be forthcoming

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, the OS returns an error message to the application trying to make the IP connection

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, the OS returns an error message to the application trying to make the IP connection

This might be “no such host” or “host unreachable”

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

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*

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 request or reply they see and modify their own ARP caches accordingly

ARP

So a gratuitous ARP would help break old associations that are no longer valid but still cached

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 cached, but now out-of-date hardware address

ARP

ARP is purely a local network thing: discover a hardware (next hop) address **on the local network**

ARP

ARP is purely a local network thing: discover a hardware (next hop) address **on the local network**

And it makes no sense for gateway to forward an ARP to another network, which might not even be of the same physical type

ARP

There is a interesting trick that shows ARP can be used for things other than it was designed to do: and shows how the Internet Protocols are incredibly *malleable*

ARP

There is a interesting trick that shows ARP can be used for things other than it was designed to do: and shows how the Internet Protocols are incredibly *malleable*

Used in the days before switches were common: this trick is unlikely to be used these days

ARP

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

ARP

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

A *bridge* is a host that joins two physical networks into one. It has two interfaces, one on each network

ARP

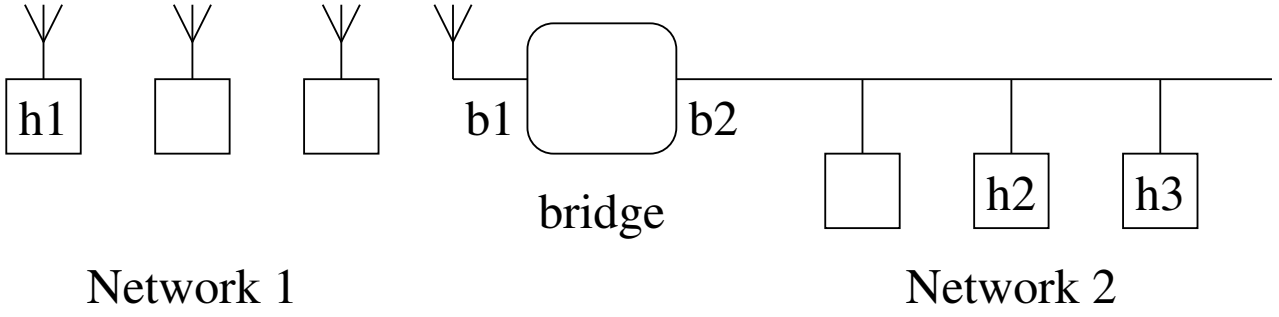
This trick allows us to extend an Ethernet (or other network) over a physically larger distance than its specifications allow, and to join a wireless network to a wired one so they appear to be a single network

A *bridge* is a host that joins two physical networks into one. It has two interfaces, one on each network

Note: this is different from a gateway we mentioned earlier, that connects two *different* networks

ARP

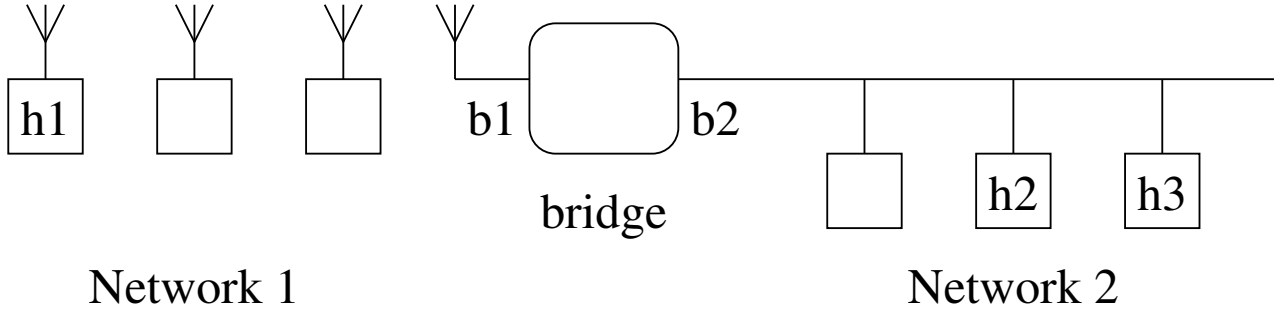
ARP Bridging



ARP bridge

ARP

ARP Bridging

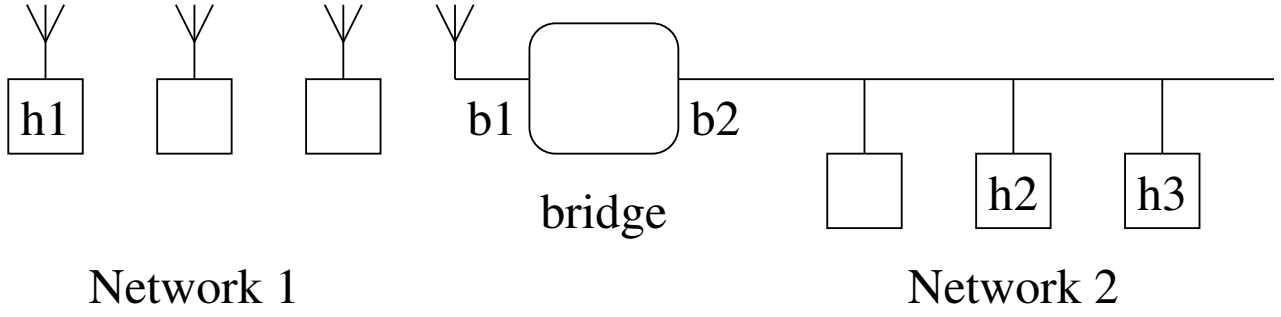


ARP bridge

This example joins a Wi-Fi to an Ethernet, but we could have any two networks that share a MAC address type

ARP

ARP Bridging



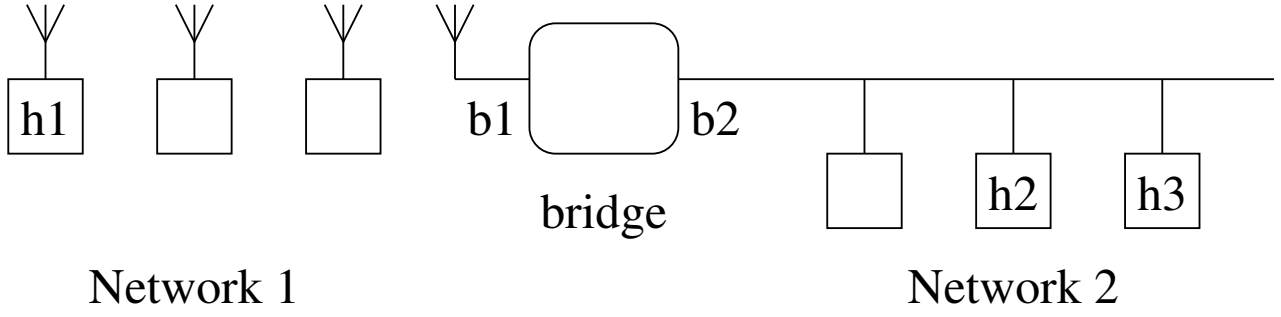
ARP bridge

This example joins a Wi-Fi to an Ethernet, but we could have any two networks that share a MAC address type

If host h1 wishes to send to host h2 it must determine its hardware address (as it is on the “same” local network)

ARP

ARP Bridging



ARP bridge

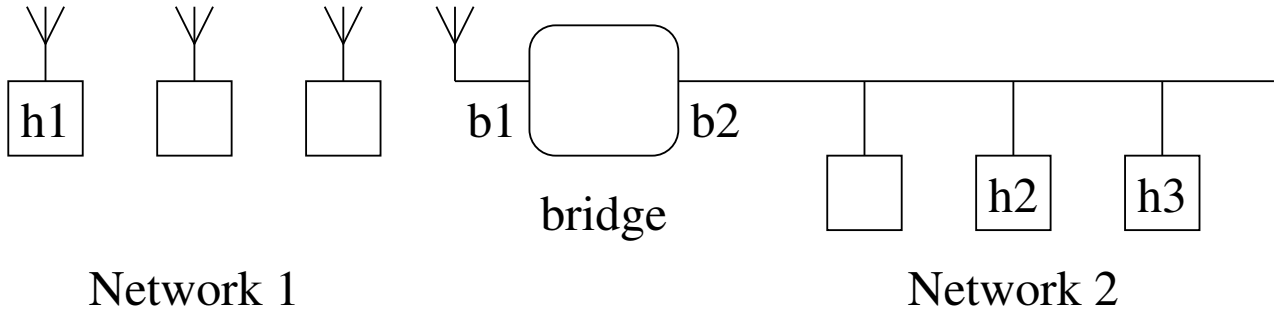
This example joins a Wi-Fi to an Ethernet, but we could have any two networks that share a MAC address type

If host h1 wishes to send to host h2 it must determine its hardware address (as it is on the “same” local network)

So h1 does an ARP broadcast for h2, just as normal

ARP

ARP Bridging



ARP bridge

This example joins a Wi-Fi to an Ethernet, but we could have any two networks that share a MAC address type

If host h1 wishes to send to host h2 it must determine its hardware address (as it is on the “same” local network)

So h1 does an ARP broadcast for h2, just as normal

The bridge sees this request and responds on behalf of h2 (a *proxy* ARP), but it supplies its *own* hardware address b1

ARP

ARP Bridging

Now h1 sends data to what it thinks is h2, but is actually the bridge

ARP

ARP Bridging

Now h1 sends data to what it thinks is h2, but is actually the bridge

The bridge reads the data packet, sees it is destined for h2 (by its IP address) and forwards it to the other network where h2 can read it

ARP

ARP Bridging

Now h1 sends data to what it thinks is h2, but is actually the bridge

The bridge reads the data packet, sees it is destined for h2 (by its IP address) and forwards it to the other network where h2 can read it

Furthermore, it rewrites the forwarded frame's header to have h2 as destination and b2 as source

ARP

ARP Bridging

Now h1 sends data to what it thinks is h2, but is actually the bridge

The bridge reads the data packet, sees it is destined for h2 (by its IP address) and forwards it to the other network where h2 can read it

Furthermore, it rewrites the forwarded frame's header 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

ARP

ARP Bridging

In either case the packet goes to the bridge, which forwards it to h1, again rewriting the frame addresses appropriately

ARP

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

ARP

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

If h1 is communicating with both h2 and h3 its cache will show them to have the *same* hardware address b1: this is not a problem

ARP

ARP Bridging

Exercise Find out if your home network does ARP bridging, or if it simply acts like a switch on a single network

Exercise Make sure you understand the difference between what a gateway does, what a switch does and what a bridge does

Virtual Bridging

Bridging is useful, but shouldn’t be taken too far

Virtual Bridging

Bridging is useful, but shouldn’t be taken too far

Larger networks have more traffic

Virtual Bridging

Bridging is useful, but shouldn't be taken too far

Larger networks have more traffic

Just think of the ARP broadcasts alone!

Virtual Bridging

Bridging is useful, but shouldn't be taken too far

Larger networks have more traffic

Just think of the ARP broadcasts alone!

It is often better to split a large network into several smaller ones: see subnetting, later

RARP

Exercise Read about *Reverse ARP* (RARP): given a hardware address find the IP address