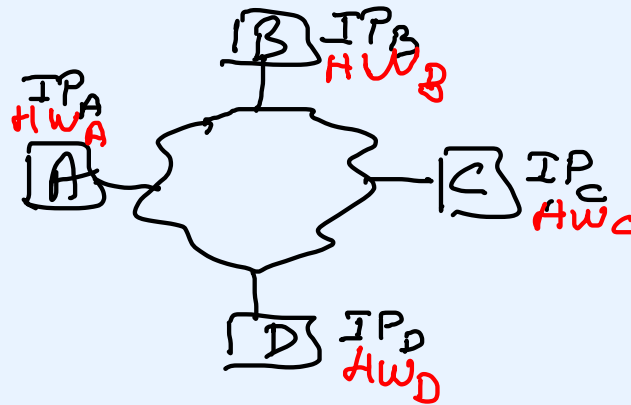


**PART V**

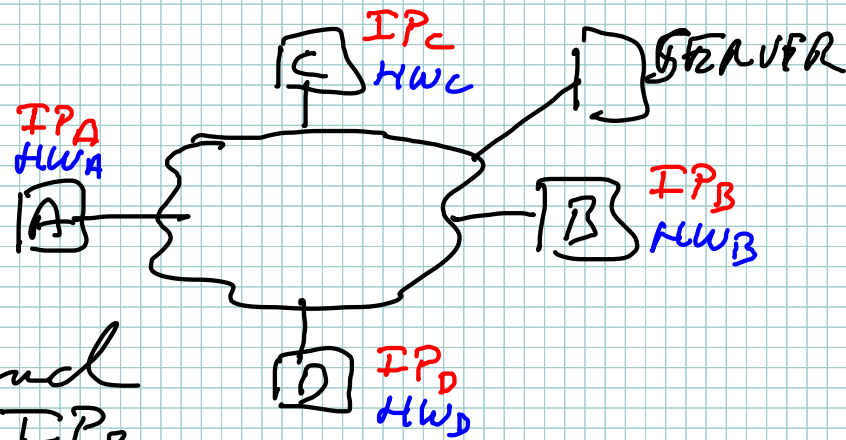
**MAPPING INTERNET ADDRESSES  
TO PHYSICAL ADDRESSES  
(ARP)**

## Motivation

- Must use hardware (physical) addresses to communicate over network
- Applications only use Internet addresses



ARP = ADDRESS RESOLUTION  
PROTOCOL  
 $IP@ \rightarrow HW@$

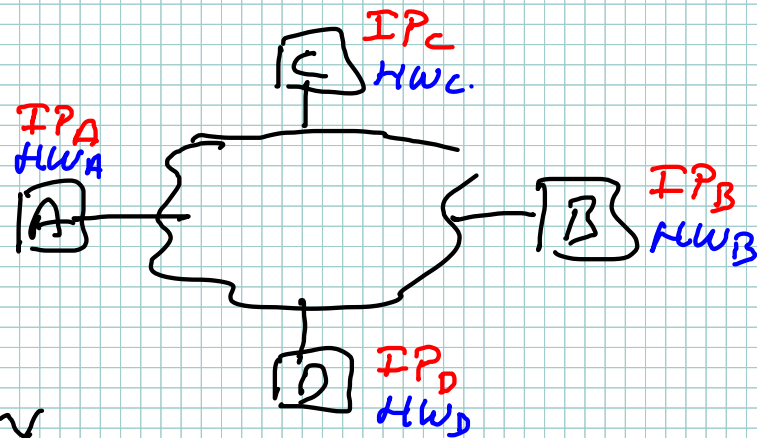


A want to send  
MSG TO IP<sub>B</sub>

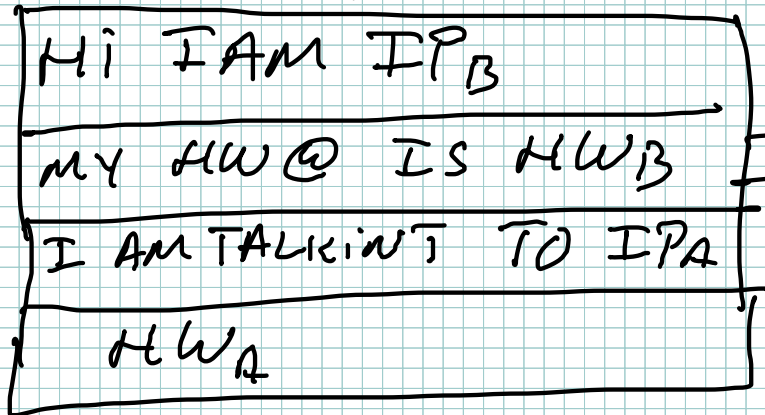
ARP REQUEST

HI I AM IP <sub>A</sub>
MY HW @ HWA
LOOKING FOR HW @ OF IP <sub>B</sub>

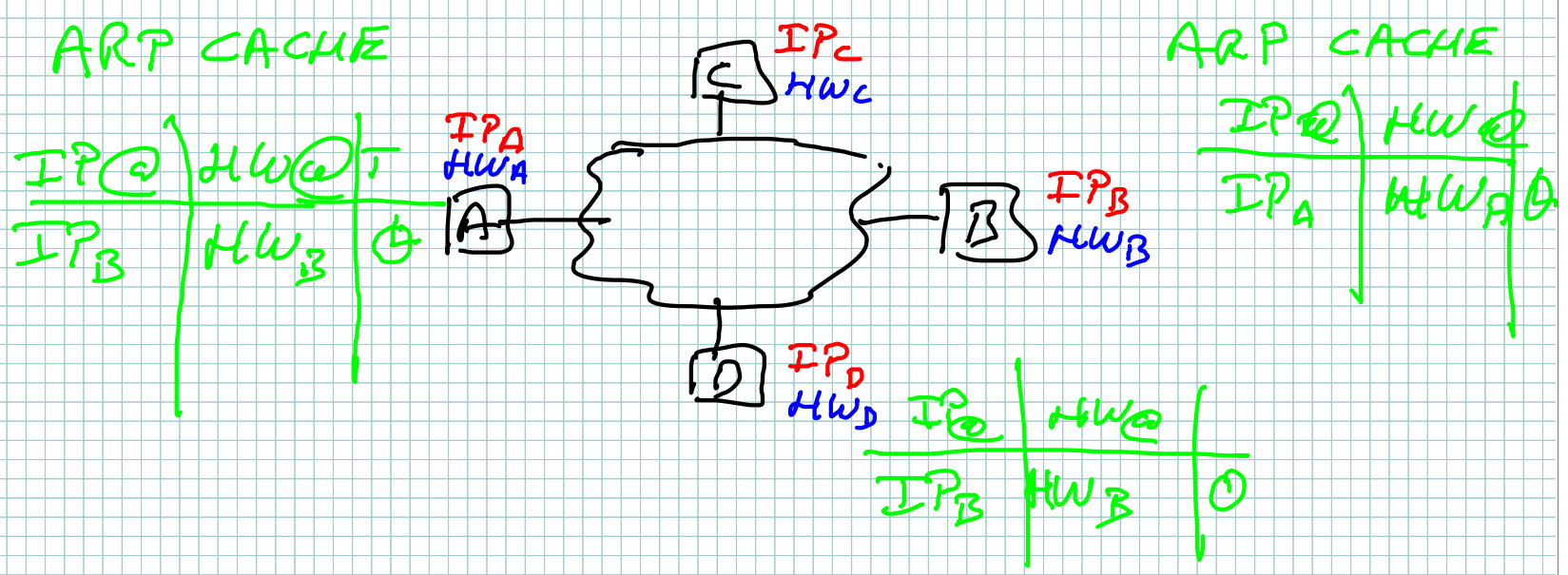
BROADCAST  
TO ALL  
ON LOCAL NETWORK



ARP REPLY



UNICAST  
FROM B → A



## Example

- Computers A and B on same network
- Application on A generates packet for application on B
- Protocol software on A must use B's hardware address when sending a packet

## Consequence

- Protocol software needs a mechanism that maps an IP address to equivalent hardware address
- Known as *address resolution* problem

## Address Resolution

- Performed at each step along path through Internet
- Two basic algorithms
  - Direct mapping
  - Dynamic binding
- Choice depends on type of hardware



## Direct Mapping

- Easy to understand
- Efficient
- Only works when hardware address is small
- Technique: assign computer an IP address that encodes the hardware address

## Example Of Direct Mapping

- Hardware: proNet ring network
- Hardware address: 8 bits
- Assume IP address 192.5.48.0 (24-bit prefix)
- Assign computer with hardware address K an IP address 192.5.48.K
- Resolving an IP address means extracting the hardware address from low-order 8 bits

## Example Of Direct Mapping

- Hardware: proNet ring network
- Hardware address: 8 bits
- Assume IP address 192.5.48.0 (24-bit prefix)
- Assign computer with hardware address K an IP address 192.5.48.K
- Resolving an IP address means extracting the hardware address from low-order 8 bits

## Dynamic Binding

- Needed when hardware addresses are large (e.g., Ethernet)
- Allows computer A to find computer B's hardware address
  - A starts with B's IP address
  - A knows B is on the local network
- Technique: broadcast query and obtain response
- Note: dynamic binding only used across one network at a time

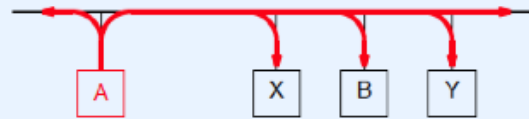
## **Internet Address Resolution Protocol (ARP)**

- Standard for dynamic address resolution in the Internet
- Requires hardware broadcast
- Intended for LAN
- Important idea: ARP only used to map addresses within a single physical network, never across multiple networks

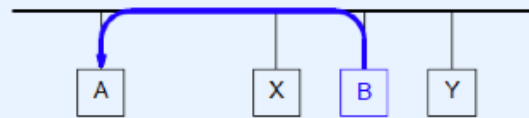
## ARP

- Machine A broadcasts ARP request with B's IP address
- All machines on local net receive broadcast
- Machine B replies with its physical address
- Machine A adds B's address information to its table
- Machine A delivers packet directly to B

## Illustration Of ARP Request And Reply Messages



A broadcasts request for B  
(across local net only)



B replies to request

## ARP Packet Format When Used With Ethernet

REQ

0	8	16	31
ETHERNET ADDRESS TYPE (1) <b>1</b>		IP ADDRESS TYPE (0800) <b>0800</b>	
ETH ADDR LEN (6) <b>6</b>	IP ADDR LEN (4) <b>4</b>	OPERATION <b>REQ</b>	
SENDER'S ETH ADDR (last 4 octets) <b>HWA</b>			
SENDER'S ETH ADDR (last 2 octets) <b>HW A</b>		SENDER'S IPA (first 2 octets) <b>IPA</b>	
SENDER'S IPA (last 2 octets) <b>IPA</b>		TARGET'S IPA (first 2 octets) <b>0000</b>	
TARGET'S ETH ADDR (last 4 octets) <b>0000 0000</b>			
TARGET'S IPB ADDR (all 4 octets) <b>IPB</b>			



## ARP Packet Format When Used With Ethernet

REPLY

0	8	16	31
ETHERNET ADDRESS TYPE (1)		IP ADDRESS TYPE (0)	
ETH ADDR LEN (6)		OPERATION (REPLY)	
SENDER'S ETH ADDR (first 4 octets)			
SENDER'S ETH ADDR (last 2 octets)		SENDER'S IP ADDR (first 2 octets)	
SENDER'S IP ADDR (last 2 octets)		TARGET'S ETH ADDR (last 2 octets)	
TARGET'S ETH ADDR (first 4 octets)			
TARGET'S IP ADDR (last 4 octets)			

## **Observations About Packet Format**

- General: can be used with
  - Arbitrary hardware address
  - Arbitrary protocol address (not just IP)
- Variable length fields (depends on type of addresses)
- Length fields allow parsing of packet by computer that does not understand the two address types

## Retention Of Bindings

- Cannot afford to send ARP request for each packet
- Solution
  - Maintain a table of bindings
- Effect
  - Use ARP one time, place results in table, and then send many packets

## ARP Caching

- ARP table is a cache
- Entries time out and are removed
- Avoids stale bindings
- Typical timeout: 20 minutes

## Algorithm For Processing ARP Requests

- Extract sender's pair, (IA, EA) and update local ARP table if it exists
- If this is a request and the target is "me"
  - Add sender's pair to ARP table if not present
  - Fill in target hardware address
  - Exchange sender and target entries
  - Set operation to *reply*
  - Send reply back to requester

## Algorithm Features

- If A ARPs B, B keeps A's information
  - B will probably send a packet to A soon
- If A ARPs B, other machines do not keep A's information
  - Avoids clogging ARP caches needlessly

## **Conceptual Purpose Of ARP**

- Isolates hardware address at low level
- Allows application programs to use IP addresses

## ARP Encapsulation

- ARP message travels in data portion of network frame
- We say ARP message is *encapsulated*



## Illustration Of ARP Encapsulation



## **Ethernet Encapsulation**

- ARP message placed in frame data area
- Data area padded with zeroes if ARP message is shorter than minimum Ethernet frame
- Ethernet type 0x0806 used for ARP

## **Reverse Address Resolution Protocol**

- Maps Ethernet address to IP address
- Same packet format as ARP
- Intended for bootstrap
  - Computer sends its Ethernet address
  - RARP server responds by sending computer's IP address
- Seldom used (replaced by DHCP)

## Summary

- Computer's IP address independent of computer's hardware address
- Applications use IP addresses
- Hardware only understands hardware addresses
- Must map from IP address to hardware address for transmission
- Two types
  - Direct mapping
  - Dynamic mapping

## **Summary (continued)**

- Address Resolution Protocol (ARP) used for dynamic address mapping
- Important for Ethernet
- Sender broadcasts ARP request, and target sends ARP reply
- ARP bindings are cached
- Reverse ARP was originally used for bootstrap