COMP 204 – Principles of Computer Networks

Week 3

FRANKLIN UNIVERSITY

Agenda

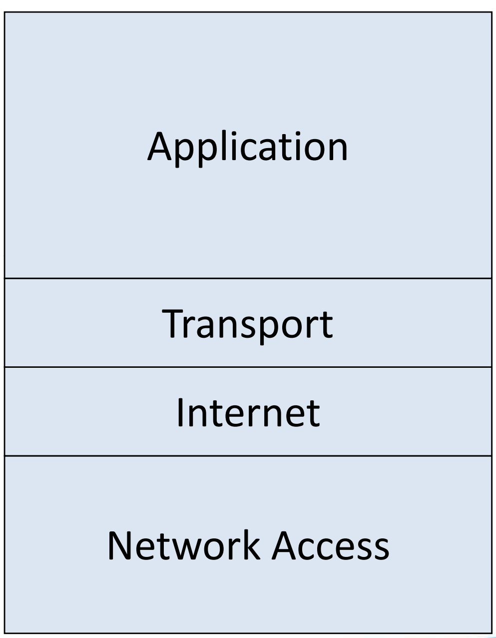
- Review this week's learning outcomes
- Presentation of this week's material
- Introduce homework problems
- Q & A session

FRANKLIN UNIVERSITY

This Week’s Outcomes

- Describe IPv4 in terms of addressing, encapsulation, and routing.
- Describe the interrelationship between IP and both TCP and UDP.
- Simulate routing between two hosts on diverse networks.

Review – Application and Transport



Review – Application and Transport

Application
Transport
Internet
Network Access

OSI layers 5, 6, 7. User applications, services, and application layer protocols. "messages"

- Ex: browser, httpd, and HTTPD working together.
- App message -> app protocol -> transport layer.

Review – Application and Transport

Application
Transport
Internet
Network Access

OSI layer 4. Typical examples are UDP and TCP. "segments"

- Port – specifies target service/app listening for messages
- UDP – "connectionless"
- TCP – "connection oriented"
- Header differences
- Handshaking in TCP

This week

Application

Transport

Internet

Network Access

How messages get from host to host. Layer 3. "packets"

- Ex: IP (Internet Protocol) and ICMP (Internet Control Message Protocol)

Chapter 5

- Internet layer (4 on OSI model)

IP – Virtual Network

- IP allows for interconnection of different hosts on the network. In order for communication to take place, a stream of data is divided into **packets** that travels along a certain **route** from the **source IP address** to **destination IP address**.

IP – Functions

- **Addressing**
  - Specifies the rules for addressing on IP, broadcasts, multicasts…
- **Encapsulation**
  - Specifies how the information about source and destination suppose to fit in the IP packet
- **Routing**
  - Specifies how the packets will flow from the source to the destination
- **Decapsulation**
  - Specifies how the encapsulated data suppose to be removed

IP – Features

- Connectionless
- Best effort (unreliable)
- Media independent

IP – Connectionless

- Does not establish connection before sending data
- Each packet is independent of the others
- Packets from the same “virtual circuit” can take different paths

IP – Best Effort Delivery

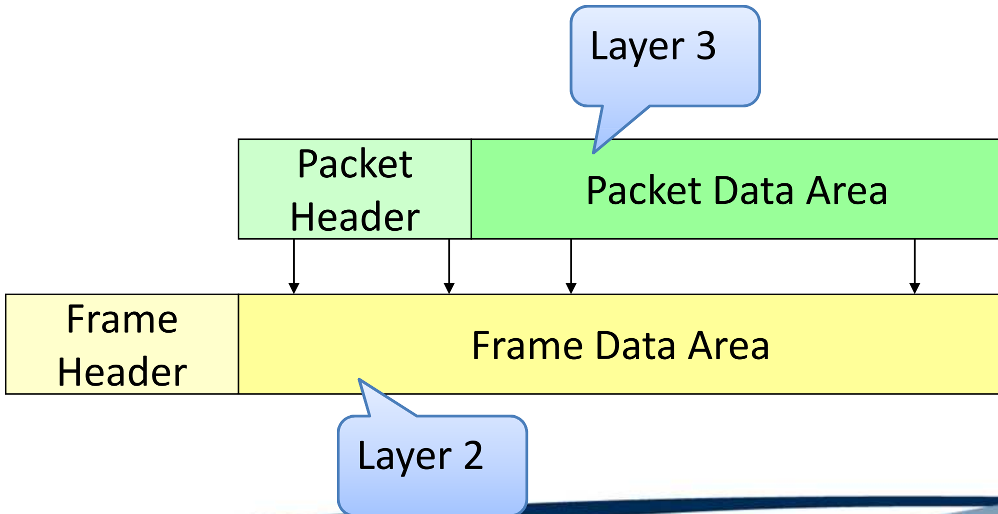
- No confirmation of delivery of the packets
- Packet may be lost, duplicated, delayed or delivered out of sequence
- There is nothing in IP to prevent or handle this
- Reliability can be achieved with TCP
- Absence of reliability makes IP very efficient. How?

IP – Media Independence

- Does not care about underlying medium, can be
  - Ethernet, Fiber, wireless, etc.
- Handles Fragmentation that arises if different medium across the packet route has different Maximum Transfer Unit (MTU)

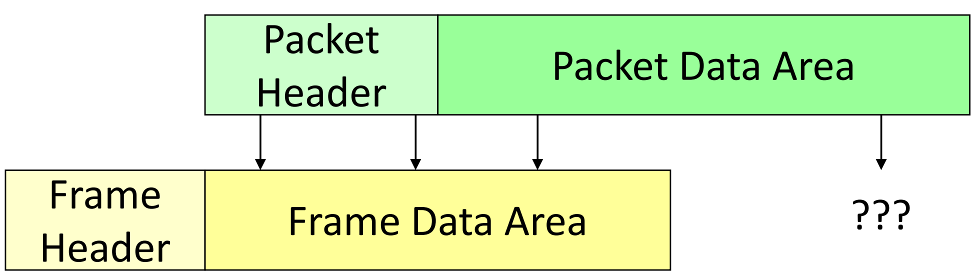
IP – Packet Fragmentation

- Packet in frame data



IP – Packet Fragmentation

- Maximum packet size is 65,535 bytes (2^{16} since we have 2 bytes to store packet length)
- What if the physical frame size is smaller than the IP Packet?



IP – Packet Fragmentation

- Maximum Transfer Unit (MTU) on Layer 2:
  - Token Ring (16 Mbps) 17,914 bytes
  - Token Ring (4 Mbps) 4,464 bytes
  - FDDI 4,352 bytes
  - Ethernet 1,500 bytes
  - Some as small as 128 bytes
- Choose least common denominator?
  - Unnecessarily inefficient
  - Will result in very small packets on networks that support large MTUs

FRANKLIN UNIVERSITY

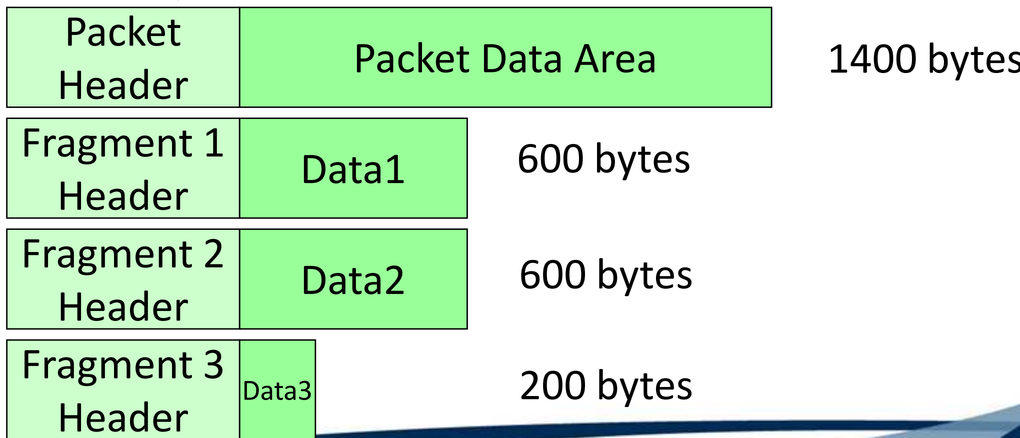
IP – Packet Fragmentation

- IP protocol:
  - Hides underlying hardware details
  - Divides large packets into smaller pieces (fragments)
  - Reassembles the fragments at destination

FRANKLIN UNIVERSITY

IP – Packet Fragmentation

- Each fragment contains header with information that is similar to the original packet except for the few fields



IP – Packet Fragmentation

- Reassembled at final destination
- Prevents packet from being fragmented and reassembled multiple times
- May cause inefficiency after fragmented
- Fragments may be lost
  – target uses reassembly timer

IP – Datagram

- Basic unit of transfer is a datagram (you will often see it referred to as "packet")
- Consists of header area and data area

IP – Datagram Structure

<--- byte 1 - - -> <--- byte 2 - - -> <--- byte 3 - - -> <--- byte 4 - - ->

<table>
<tr><td>Version</td><td>Hdr Len</td><td colspan="2">Service Type</td><td colspan="4">Packet Total Length</td></tr>
<tr><td colspan="4">Identification</td><td>Flags</td><td colspan="3">Fragment Offset</td></tr>
<tr><td colspan="2">Time to Live</td><td colspan="2">Protocol</td><td colspan="4">Header Checksum</td></tr>
<tr><td colspan="8">Source IP Address</td></tr>
<tr><td colspan="8">Destination IP Address</td></tr>
<tr><td colspan="6">IP Options (if any)</td><td colspan="2">Padding</td></tr>
<tr><td colspan="8">Data</td></tr>
<tr><td colspan="8">. . .</td></tr>
</table>

IP – Version

Version	Hdr Len	Service Type	Total Length

- 4 bits
- Version of IP Protocol which created packet
- Ensures both sender and receiver interpret remainder of header the same
- Current IP protocol version is 4, can be 6 for IPv6

FRANKLIN UNIVERSITY

IP – Header Length

Version	**Hdr Len**	Service Type	Total Length

- 4 bits
- Length of the entire header in 32-bit words (this allows the header field to be smaller)
- Actual header length may vary depending on options (see later)
- Most common value (no options) is 5 (20 bytes)

FRANKLIN UNIVERSITY

IP – Service Type

Version	Hdr Len	**Service Type**	Total Length

Precedence	D	T	R	C	0

- 8 bits or 1 byte
- **Precedence** – 3 bits
  - 0 - normal, 7 - network control
  - could be used in congested networks
- Flags – 4 bits:
  - D – Minimize delay (real-time communication, Telnet)
  - T – Maximize throughput (bulk data transfer, SMTP)
  - R – Maximize reliability (important communication, SNMP)
  - C – Minimize cost
- Not a guarantee, but hints to routers

IP – Total Length

Version	Hdr Len	Service Type	**Total Length**

- 16 bits or 2 bytes
- Length of the entire IP packet in bytes
- Includes header and data
- **Length of data** is **total length** - **header length**
- Maximum length 2^{16}-1 or 65,535

IP – Fragmentation – Identification

Identification	Flags	Fragment Offset

- 16 bits
- Unique integer that identifies the datagram
  - Unique when combined with source address
  - Usually a counter
  - All fragments of the original datagram share the same Identification Number
  - Helps in reassembly of fragments

FRANKLIN UNIVERSITY

IP – Fragmentation – Flags

Identification	**Flags**	Fragment Offset

- Flags (3 bits)
  - One unused
  - Do not fragment bit
    - Request not to fragment this packet
    - If a datagram cannot be sent unfragmented, the ICMP error is sent to the source
  - More fragment bit
    - Set for all but last fragment
    - Used to find out when the last fragment is received

FRANKLIN UNIVERSITY

IP – Fragmentation – Fragment Offset

Identification	Flags	**Fragment Offset**

- 13 bits
- Offset in original datagram where fragment begins
- Measured in units of 8 bits
- Starts at zero

IP – Fragmentation Control – Length

Version	Hdr Len	Service Type	**Total Length**

- Total Length (16 bits)
  - Remember that one?
  - In reality it is the length of the fragment, not the original datagram
  - The length of the original unfragmented datagram is offset of the last fragment + total length of last fragment

IP – Time To Live

Time to Live	Protocol	Header Checksum

- 8 bits
- Originally meant to hold a time stamp, which each router would decrement by the time the datagram spend. Difficult to implement – must synchronize clocks and keep the information about each datagram
- Modern approach:
  1. Store the maximum number of hops that the datagram can make
  2. Each router decrements the TTL by one
  3. When the TTL is equal to zero, router discards the datagram and ICMP error is sent to source
- Guarantees packets don't loop forever if routing tables are corrupt

IP – Protocol

Time to Live	**Protocol**	Header Checksum

- 8 bits
- Which high-level protocol created the message
- Used to interpret format of data
- Examples:
  - 1 – ICMP
  - 6 – TCP
  - 17 – UDP

IP – Header Checksum

Time to Live	Protocol	**Header Checksum**

- 16 bits
- Validates integrity of header
- Treat header as a sequence of 16 bit integers
- Add using 1's complement
- Does not check the actual data of the packet

FRANKLIN UNIVERSITY

IP – Addresses

Source IP Address
Destination IP Address

- 32 bit
- Original source and target addresses
- Never changes

FRANKLIN UNIVERSITY

IP – Options

IP Options (if any)	Padding

- Optional
- Can be for the following purposes:
  - Testing / Debugging
  - Packet or Network control
- Variable length – depends on options
- String of bytes with no separators

FRANKLIN UNIVERSITY

IP – Summary

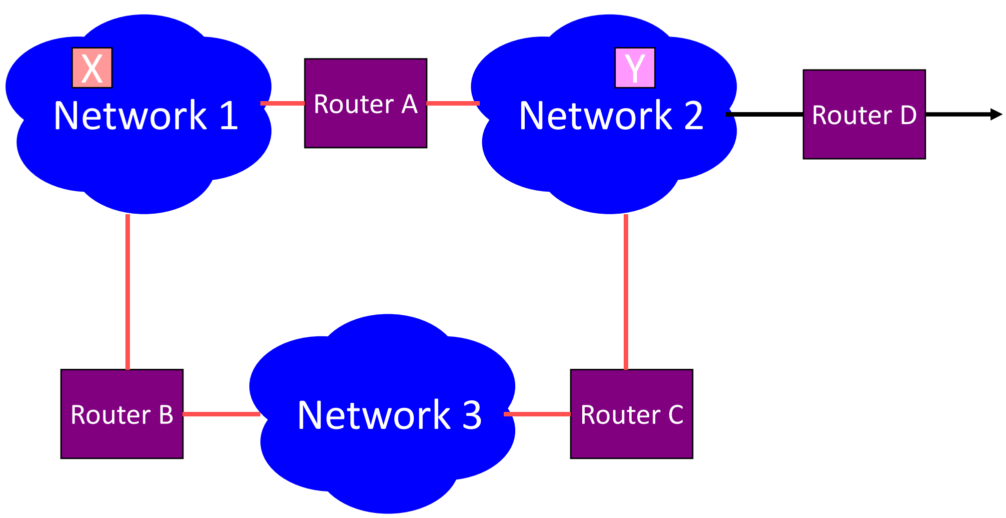
- Used to route datagrams from one host to another
- Handles fragmentation over diverse Layer 2 networks
- Best-effort delivery system – not guaranteed
- Error reporting via ICMP (Internet Control Message Protocol)
- Ability to add options

FRANKLIN UNIVERSITY

Routing – IP Datagrams

- Routing
  - Routing chooses path over which the packet will travel
  - Store and forward packet
  - Can occur within a network
  - Can occur on the end host or the router

Routing – Connecting Networks



Routing – Direct Delivery

- Direct delivery
  - Datagram goes from one machine across physical network directly to another
  - Both machines must be physically attached to same network
  - Always final step of a datagram route (may be from a router to target)

Routing – Direct Delivery

- No routers involved
- Source sends packet to target
  - Encapsulates packet in physical frame
  - Maps target IP address to physical address via ARP
  - Transmits frame to target over Layer 2 network, such as Ethernet

Routing – Direct Delivery

- How does source know if target is on same physical address?
  - Compares own network id with target network id

Routing – Indirect Delivery

- Indirect Delivery
  - Target is not on same network as source
  - Datagram must be passed to a router for forwarding
  - Source must determine an acceptable router for datagram to reach the target
  - Each router selects next router until target is reached or direct delivery can be used

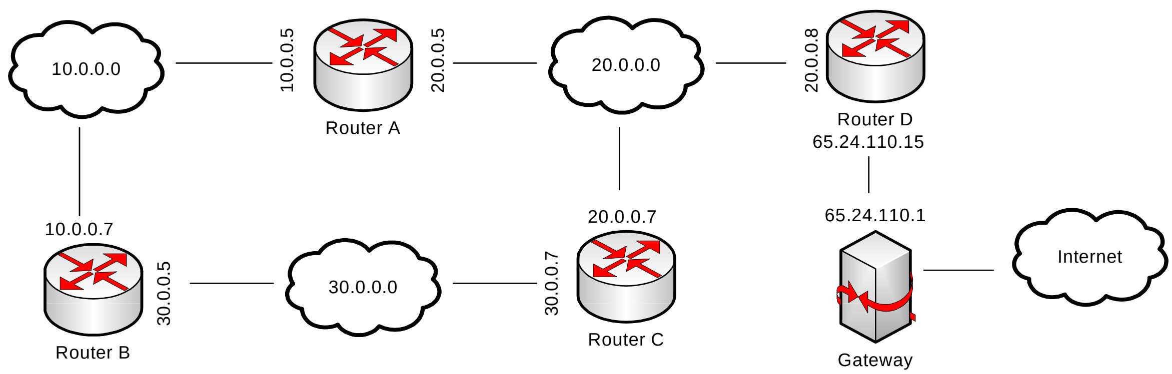
Routing – Tables

- Routing usually involves special routing table
- Both hosts and routers can have routing tables
- Table cannot contain every host in world
  - too big
  - could not keep current

Routing – Next-Hop

- Better solution – next-hop routing:
  - Table only points to next router
  - Does not know complete route
  - Contains pairs (N,R)
    - N - network ID
    - R - router IP address (only routers that are directly accessible to the host)
  - Stores the information that is needed to do the next hop – forward the datagram to the next router on the path to the target

Routing – Routing Table Example



Network	Router A	Router B	Router C	Router D
10.0.0.0	direct	direct	20.0.0.5 30.0.0.5	20.0.0.5 20.0.0.7
20.0.0.0	direct	10.0.0.5 30.0.0.7	direct	direct
30.0.0.0	20.0.0.7 10.0.0.7	direct	direct	20.0.0.5 20.0.0.7
0.0.0.0 (default)	20.0.0.8	10.0.0.5 30.0.0.7	20.0.0.8	65.24.110.1

Routing – Host Specific Routes

- Most routing software allows per-host routes
- Allows administrator more control
- Allows testing specific routers
- Can be used for security purposes

Routing – Default Routes

- If target network is not in routing table send to default router
- Using previous example
  - Network 40 would go through default router D (20.0.0.8)
- Works well if only one router attached to outside world

Routing –Algorithm

- RouteDatagram(Datagram, Routing Table)
  1. Extract target IP address, D
  2. Calculate network prefix, N from D
  3. If delivering directly, send over local network
  4. Else if host specific entry, send to next-hop
  5. Else if network prefix entry, send to next-hop
  6. Else if default route entry, send to next-hop
  7. Else declare routing error

Routing – Routing with IP Addresses

- Routing table contains IP address
- Must be translated to hardware address before sending datagram
- All routers in table must be local
- Why not use hardware address in table?

FRANKLIN UNIVERSITY

Routing – Routing with IP Addresses

- Why use IP addresses in routing table?
  - IP addresses easier for administrator to manage
  - IP layer hides hardware details
  - hardware address differ in size and form
  - software is layered; easier to test and modify

FRANKLIN UNIVERSITY

Routing – Receiving Datagrams

- Router
  - Checks IP address to see if datagram has reached final destination (router management datagrams)
  - Decrements Time To Live
    - Discard datagram if zero
  - Does not otherwise change datagram
  - Forwards datagram to next hop

Routing – Receiving Datagrams

- Hosts
  - Checks IP address to see if datagram has reached final destination
  - Passes datagram to appropriate layer above
  - Discards if not final destination
  - Should not forward mis-delivered datagrams

Routing – Routing Tables

- How are they established?
  - Routers share information with each other
- Examples:
  - IP based: OSPF, IGRP, EIGRP
  - UPD based: RIP
  - TCP based: BGP

FRANKLIN UNIVERSITY

Routing – Routing Strategies

- Static
  - routes computed once and never change
- Dynamic
  - routes computed initially but change as network conditions change

FRANKLIN UNIVERSITY

Routing – Routing utilities

- `traceroute` (`tracert` on Windows)
  - – Traces route to target host
  - – Sets Time To Live to one
    - receives error and displays information
    - increase TTL and tries again

Routing – tracert example

```
C:\WINDOWS\system32\cmd.exe

C:\Documents and Settings\whittakt>tracert www.yahoo.com

Tracing route to any-fp.wa1.b.yahoo.com [67.195.160.76]
over a maximum of 30 hops:

  1     1 ms     1 ms     3 ms  10.1.22.2
  2     3 ms     1 ms     3 ms  64.129.112.1
  3     2 ms     1 ms     1 ms  66-194-130-169.static.twtelecom.net [66.194.130.169]
  4    13 ms    13 ms    14 ms  ash1-pr2-xe-1-3-0-0.us.twtelecom.net [66.192.244.70]
  5    14 ms    27 ms    38 ms  ae-6.pat1.dce.yahoo.com [216.115.102.172]
  6    13 ms    16 ms    16 ms  xe-7-0-0.msr1.ac2.yahoo.com [216.115.108.125]
  7    15 ms    14 ms    16 ms  xe-4-2-0.clr4.ac4.yahoo.com [76.13.0.103]
  8    22 ms    20 ms    16 ms  ge-0-0-0.clr2.ac4.yahoo.com [76.13.0.29]
  9    15 ms    15 ms    16 ms  ir1.fp.vip.ac4.yahoo.com [67.195.160.76]

Trace complete.

C:\Documents and Settings\whittakt>_
```

Routing – Routing utilities

- `netstat`
  - shows network status
  - "`-r`" - display routing table
- `route`
  - manipulates the routing tables

This Week's Outcomes

- Describe IPv4 in terms of addressing, encapsulation, and routing.
- Describe the interrelationship between IP and both TCP and UDP.
- Simulate routing between two hosts on diverse networks.

Self Quiz

- What layer is IP in the OSI model?
- What does IP "connect"
- How do packets move on the same network?
- How do packets move between networks?
- At what layer of OSI do routers operate?
- What is the algorithm for routing?

FRANKLIN UNIVERSITY

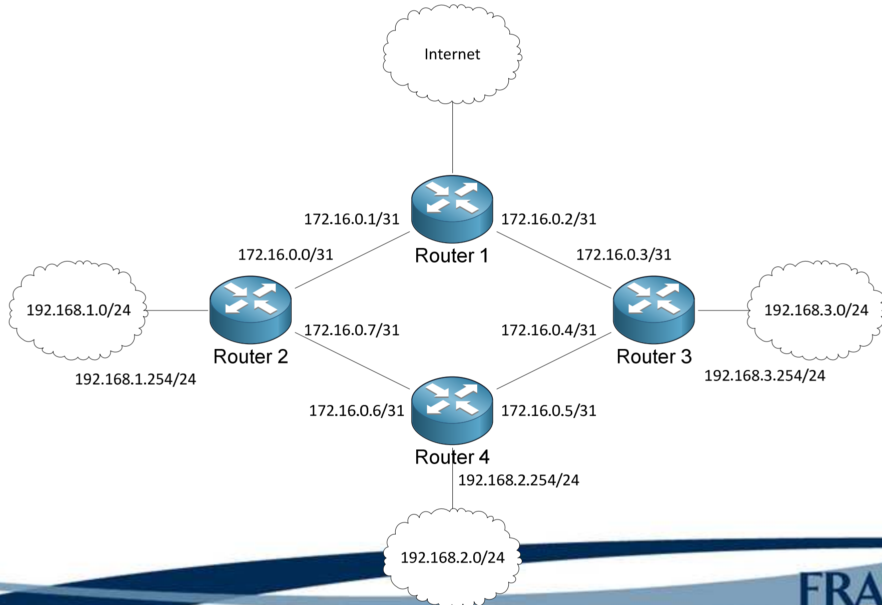
Self Quiz

- What does a router do if it receives a packet and there is no route for it?
- What happens if a packet (layer 3) is too big to fit in a frame (layer 2)?
- What does a router do with a packet destined for a host on directly connected network? An indirectly connected network? A network that it knows nothing about?

FRANKLIN UNIVERSITY

Self Quiz

- What are each of the routing tables?



Due this week

- Homework 2
- Participation 3

Next week

- Chapter 6 – how to partition networks effectively.

Q & A

- Questions, comments, concerns?