MANCHESTER 1824
The University of Manchester

COMP28411 Computer Networks

Nick Filer

Multimedia – 4

Some material from:

Kurose & Rose – Chapter 7 + Slides

Halsall – Multimedia Communications

Repeated from Multi-Media 3 Notes

MANCHESTER 1824
The University of Manchester

Problem ????

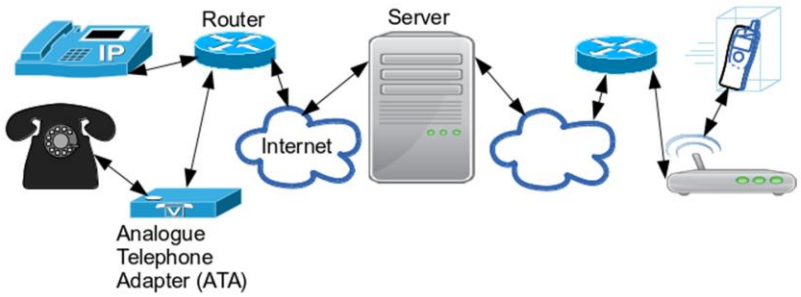
REAL-TIME CONVERSATIONS?

- Setup conference sessions
  - Session Initialization Protocol (SIP)
- Real Time Protocol (RTP) &
  - Real Time Control Protocol (RTCP)
- Making TCP work for streaming and real-time

How do today's MM sessions work?

Repeated from Multi-Media 3 Notes

- SIP – Session Initiation Protocol
  - Inter-Asterisk-eXchange (IAX) or now IAX2 competes with SIP. IAX2 only uses 1 port, less bandwidth for control, binary not text. But it is not yet fully standard.
  - Application layer , not a service provider, other protocols provide services e.g. Real Time Protocol (RTP) to carry MM traffic.
  - Uses a client server model.

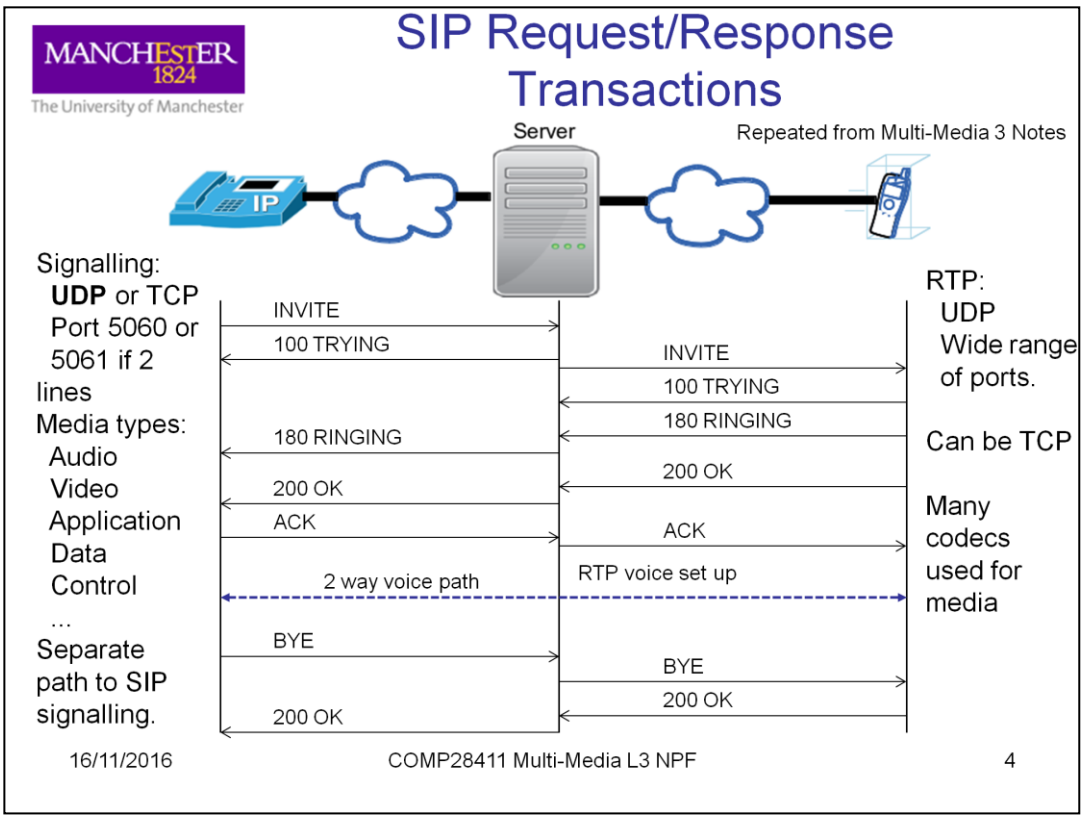


16/11/2016 COMP28411 Multi-Media L3 NPF 3

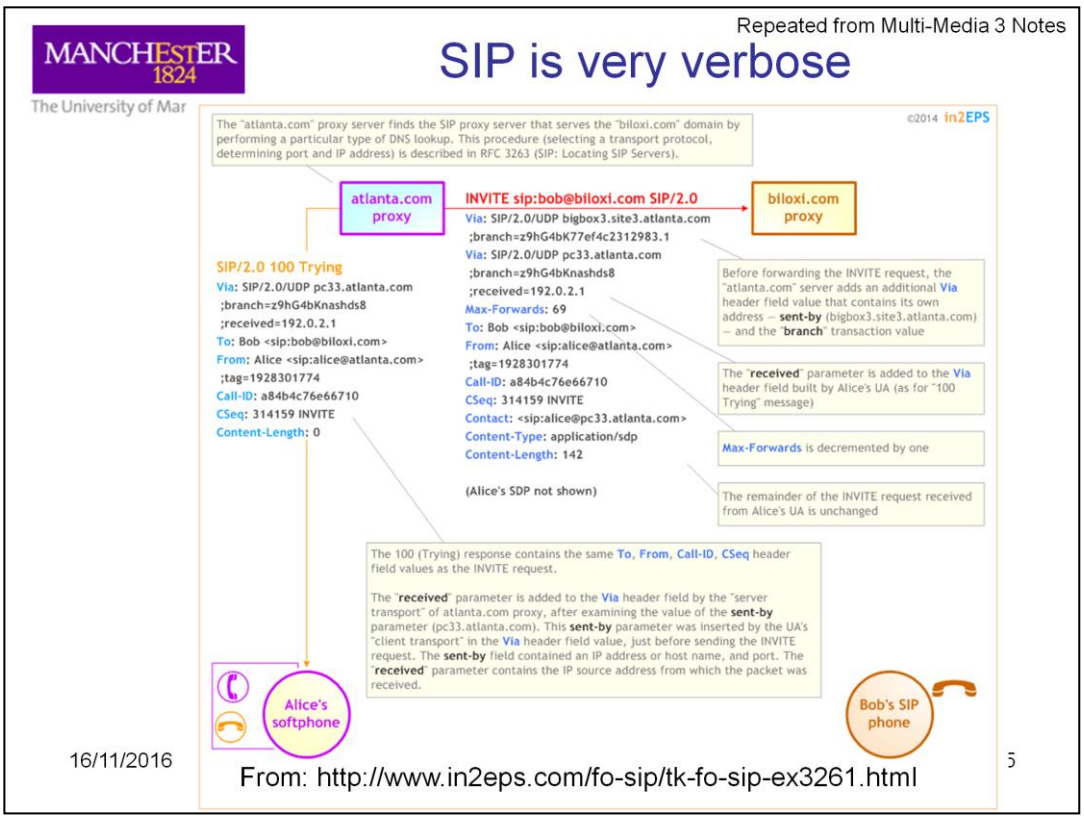
SIP can be used for one way streaming or for two way multi-media conversations. There can be 2 or more participants and there may be many separate streams of media carrying, for example voice and pictures together or separately between participants. SIP does not handle the actual media, just the configuration of the links.

SIP is a client server architecture. Each participant connects via a server and destinations are selected by SIP servers talking to one another. Hence SIP addresses normally explicitly state the name of the destination server and end up looking like email addresses. My SIP address could be nickfiler@some.sip.server.com or similar. There are quite a lot of different SIP providers.

The main competition for SIP has been from IAX and IAX2 which are part of the open-source Asterisk (http://www.asterisk.org/) system. But even Asterisk now promotes SIP.



SIP runs in real-time by passing messages between devices. The slide shows a sequence that abstractly represent a telephone conversation over SIP. Of course, SIP can handle both telephone and VoIP and other media conversations. The messages are a mixture of commands and status responses. As you might expect there are many different status responses to handle all the different outcomes and problems the media exchange system can have.



The example shows Alice trying to call Bob's SIP phone. Note that what might be a simple message is expanded with lots of fields which are necessary to uniquely configure, maintain and charge for the call!

I have deliberately ignored the SDP mentioned in this slide so far! SDP is Session Description Protocol which is solely concerned with the media for the SIP session. It works by offering a whole load of options to the remote end (this is what I can send) and letting the remote end pick a protocol and codec to use.

There are 3 main sections, for configuring the session, its timing and the media. It consists of lots of text lines of the form "field = value". Fields give the session type, the originators network location, time + time zone adjustments, encryption key, bandwidth data, list of media protocols and for each a codec. It is actually very simple. For more on this and a useful example go to: http://tinyurl.com/jarq6u5 or google "sip sdp"..

Repeated from Multi-Media 3 Notes

MANCHESTER 1824
The University of Manchester

SIP Response Codes - 1

- 1xx – Informational
  - 100 trying
  - 180 Ringing
  - 181 Call is being forwarded
  - 182 Queued
  - 183 Session progress
- 2xx – Success
  - 200 OK
  - 202 Accepted: Used for referrals
- 3xx Redirection
  - 300 Multiple choices
  - 301 Moved permanently
  - 302 Moved temporarily
  - 305 Use proxy
  - 380 Alternative service
- 4xx – Failure
  - 400 bad request
  - 401 Unauthorized
  - 402 Payment required
  - 403 Forbidden
  - 404 Not found
  - 405 Method not allowed
  - 406 Not accepted
  - 407 Proxy authentication required
  - 408 Request timeout
  - 410 Gone
  - 413 Request entity too large
  - 414 Request URI too long
  - 415 Unsupported media type
  - 416 Unsupported URI scheme

16/11/2016 COMP28411 Multi-Media L3 NPF 6

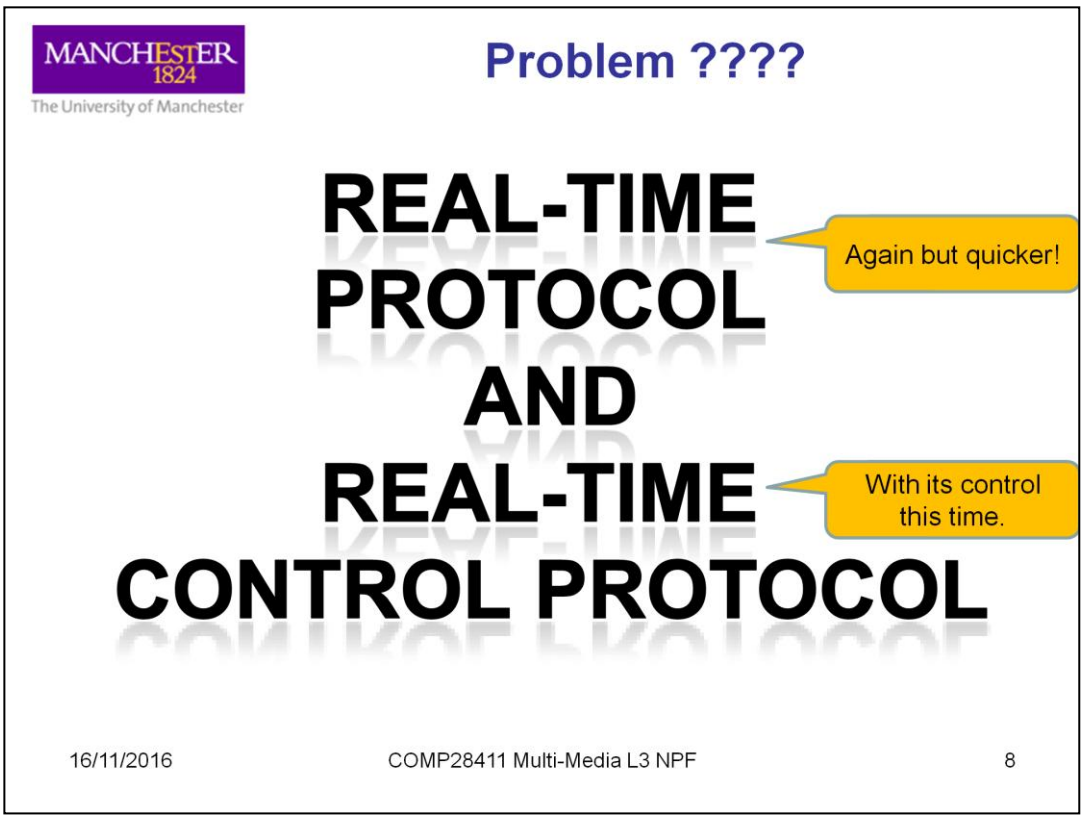
As I said there are lots and lots of SIP response codes to handle common interactions and problems in setting up conversations, maintaining them and later tearing them down.

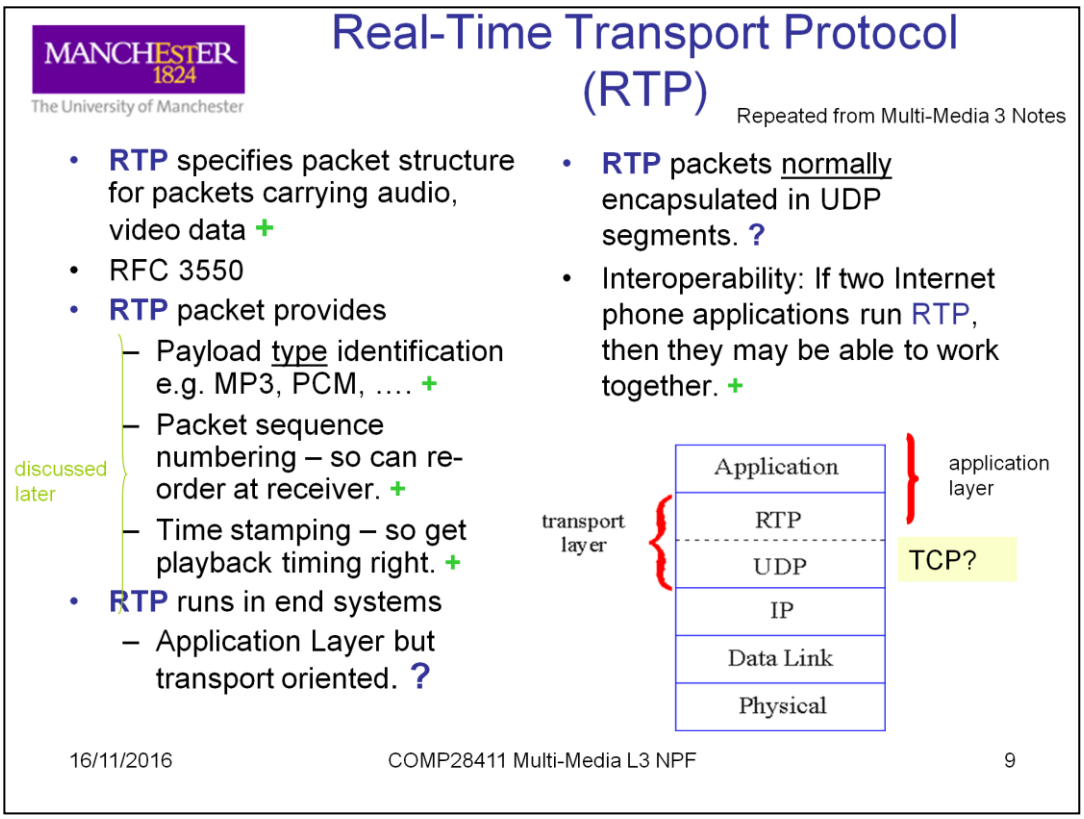
Remember that SIP runs in real-time. It has quite strict timing requirements. Messages must arrive within the allowed delays. Otherwise, instead of getting one of the success codes, one of the failure codes is sent instead. Then some form of recovery, such as starting again, must take place.

Repeated from Multi-Media 3 Notes

SIP Response Codes - 2

- 4xx – Failure (continued)
  - 420 Bad extension
  - 421 Extension required
  - 423 Interval too brief
  - 480 Temporarily unavailable
  - 481 Call/transaction does not exist
  - 482 Loop detected
  - 483 Too many hops
  - 484 Address incomplete
  - 485 Ambiguous
  - 486 Busy here
  - 487 Request terminated
  - 488 Not accepted here
  - 491 Request pending
  - 493 Undecipherable
- 5xx - Server errors
  - 500 Server internal error
  - 501 Not implemented
  - 502 Bad gateway
  - 503 Service unavailable
  - 504 Server timeout
  - 505 Version not supported
  - 513 Message too large
- 6xx Global failures
  - 600 Busy everywhere
  - 603 decline
  - 604 Does not exist anywhere
  - 606 Not acceptable

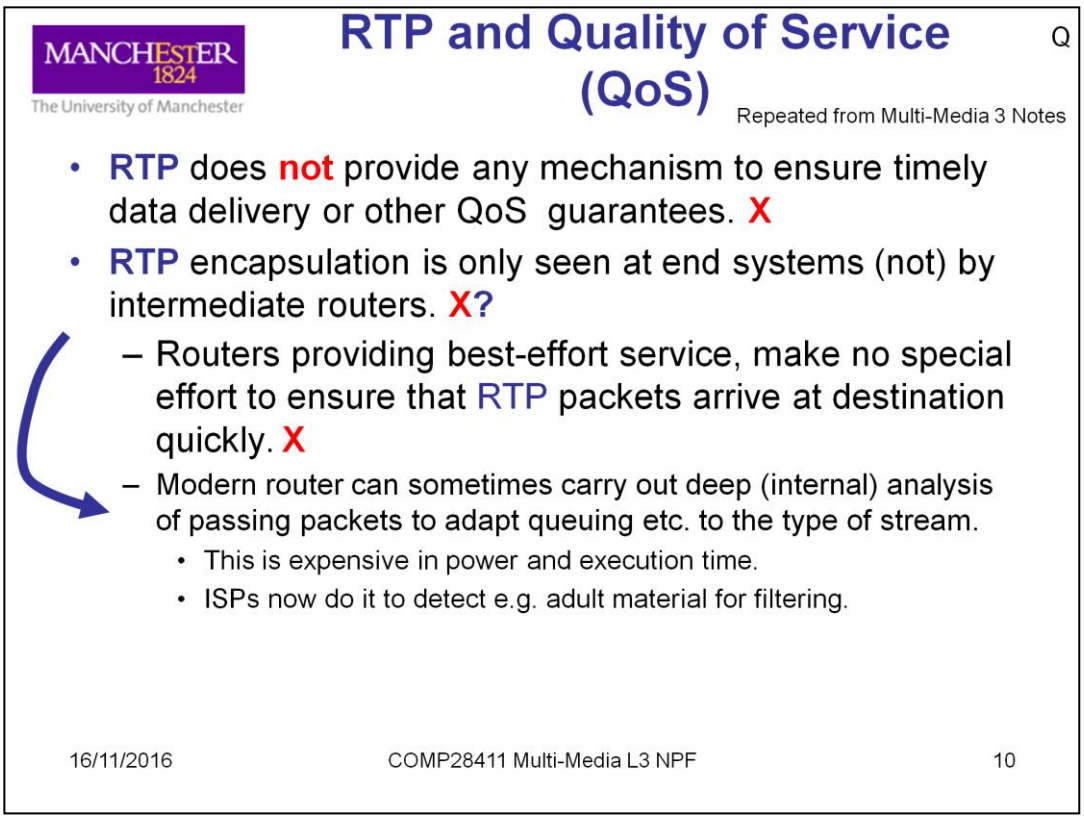




We have no doubt mentioned the RTP several times already. RTP is a protocol used to extend, in particular UDP and similar transport protocols to carry real-time media in a form whereby it can be, at the destination and provided it arrives on time, played back in the correct sequence with the correct inter packet timing so that a media stream sounds and looks OK. In addition, the protocol allows the exact format of the media data to be specified so that the destination can correctly interpret it, even if it changes from packet to packet.

Typically, destinations open a port for receiving incoming RTP packets. In a conference environment or where several applications are all generating RTP at the same time there can be multiple streams of media arriving and being processed at the same time. This requires a per media stream index as an addition field in each packet to indicate the source from where this packet stream arrived so that it can be processed correctly relative to the packets just before and after it from the same stream and from other streams.

For example, you might be in a video conference with several people. From each person's computer you receive an audio stream and a video stream separately. The packet sequence numbers and timestamps are not enough to play the data back correctly. Each stream needs to be processed separately then combined for playback.



The Real Time Protocol (RTP) is much more limited in capability than you might at first expect. However, other protocols can be used alongside RTP to expand and control what it is able to do and be used for. Firstly RTP carries timestamps from place to place associated with media objects to which the timestamp applies but it does not do anything about making sure the media is delivered on time. Partly this is because RTP is an application layer protocol which used end-to-end transport such as UDP or TCP to move packets of data from source to destinations. In theory routers are unaware of RTP but modern routers using deep packet analysis may be able to differentiate RTP packets from other protocols and types. You have seen this type of filtering in the laboratory using Wireshark.

Repeated from Multi-Media 3 Notes

MANCHESTER 1824
The University of Manchester

RTP Header

RTP Header

Payload Type (7 bits): Indicates type of encoding currently being used. If sender changes encoding in middle of conference, sender informs receiver via payload type field. For example:

- Payload type 0: PCM mu-law, 64 kbps
- Payload type 3, GSM, 13 kbps
- Payload type 7, LPC, 2.4 kbps
- Payload type 26, Motion JPEG
- Payload type 31. H.261
- Payload type 33, MPEG2 video

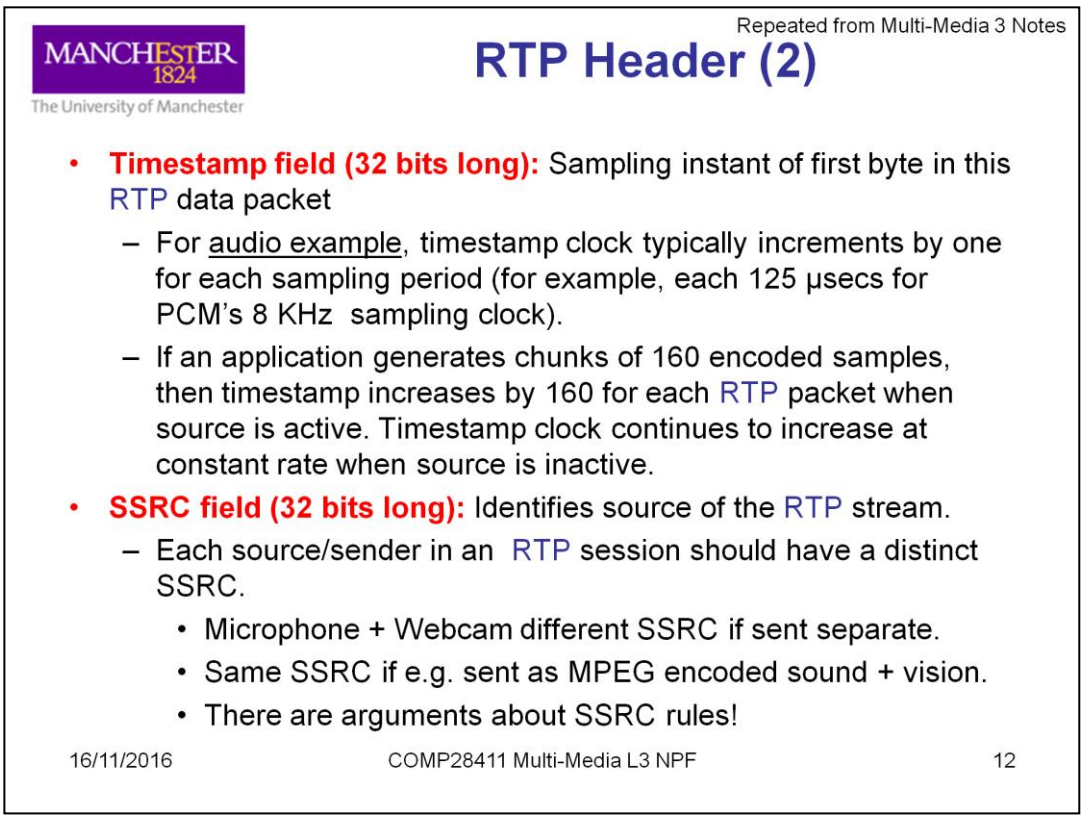
Sequence Number (16 bits): Increments by one for each RTP packet sent, and may be used to detect packet loss and to restore packet sequence. May cycle quite quickly – issue?

16/11/2016 COMP28411 Multi-Media L3 NPF 11

An RTP packet wraps around a block of media data which is then sent for transport across a network. The header part of the wrapper has various fields. The fields like many protocols have some pre-defined fixed values but in many cases lots of undefined values are left that users might grab and use for their own, in the case of the payload type, codecs and data formats.

The sequence number is 16 bits and is used as a packet counter with 64K values. It is used to ensure that packets arriving at a destination can be put back into the order they were sent if they arrive out of order. This is similar to the sequence number in TCP and other protocols. RTP could use TCP for transport but it can also be used with protocols such as the much simpler UDP that has no sequence number. In order play-back is very important to media so a field is used for this ordering information.

The small size of the sequence counter is possibly a problem. If the jitter on a network is large and the network is transferring very large numbers of RTP packets in a very short time then the cycle time for sequence numbers may be too fast such that two packets with the same sequence number are in the network at the same time. So far, this has not been a major issue. Because, in theory a UDP packet can carry 64K of data in one chunk the sequence number problem could be solved by using larger packets. However, because the ubiquitous Ethernets allow only small MTU values (see TCP notes) such large packets may end up being fragmented. Hence most media packets are relatively small and in some cases very small which is an efficiency issue because lots of packets each carrying their header and tail data and tiny amounts of actual data are much less efficient resource users than fewer large packets such as typical TCP ones fitted to the common Ethernet MTU of 1500 bytes/Octets.

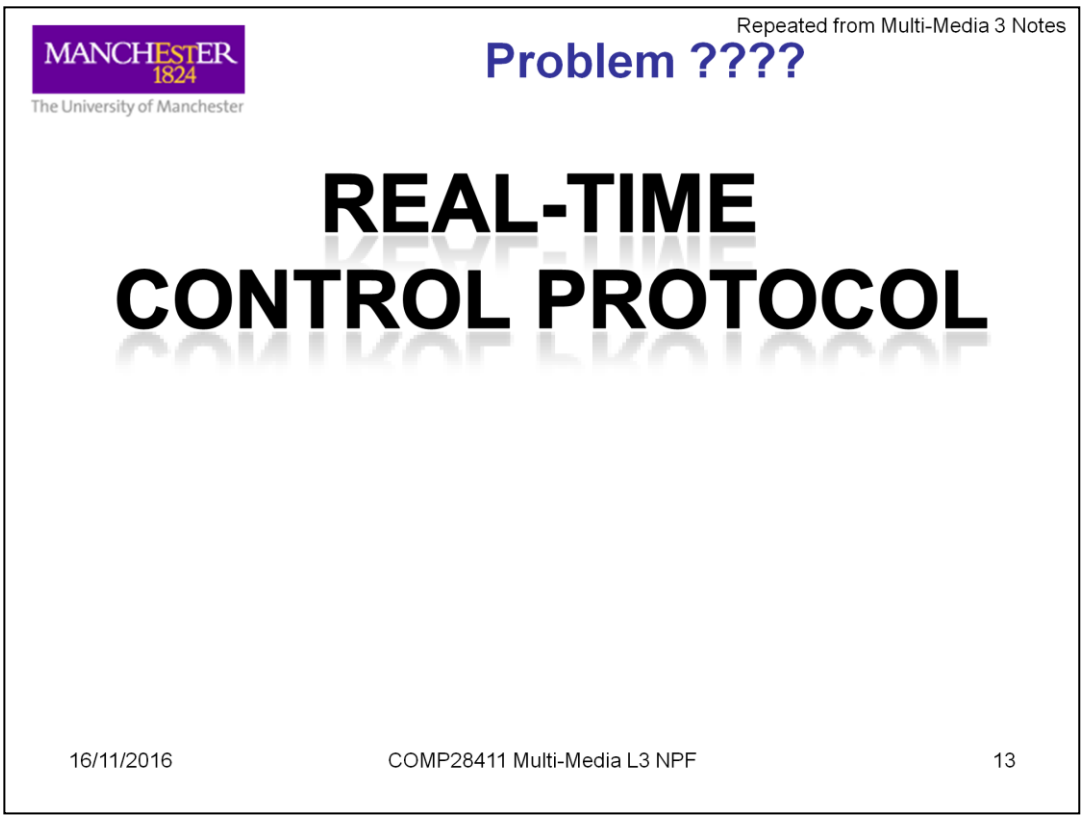


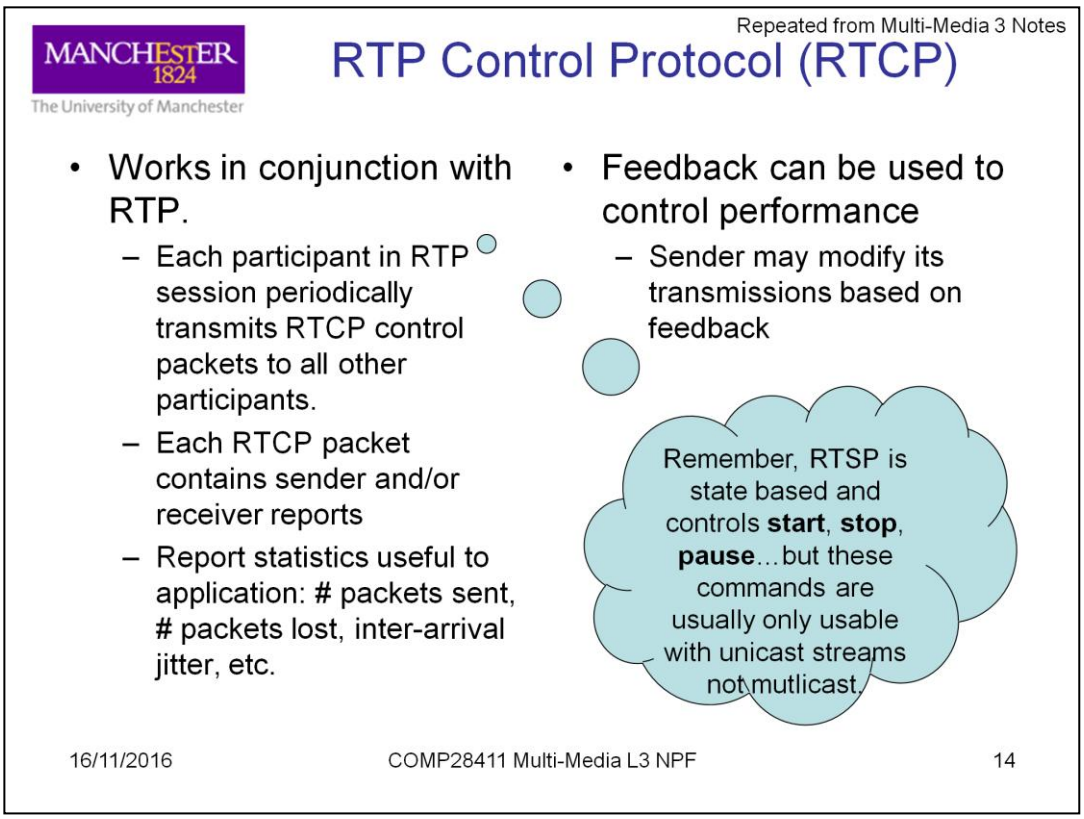
The timestamp is recorded at the source as the first sample (magnitude of sound, frame of video ….) sent in the packet is collected by the source device. Using the audio G711 CODEC 8,000 samples per second are taken at 125μsecs intervals. Note a μsec is 1 millionth of a second or 10^{-6}. The timestamp is another counter that increments in sample time units whether data is being collected or not. Therefore, it accurately reflects the separation in time between samples in adjacent or distantly separated (in time) packets. This separation in time allows the play-back to align the samples exactly the same distance apart in time for play back as they were when originally captured.

Because RTP sessions may have many different streams of media the final field discussed is an identifier for these streams. For example, imagine a studio recording with 32 different tracks or musical instruments to be merged into a final audio recording for sale to the public, each track may have an individual identifier to allow the destination to accurately reconstruct the original data. Destinations may receive data from many locations and with many sources and in different formats all at the same time. The SSRC and the Payload Type allow these various sources to be properly processed by the destination.

.

SSRC – Synchronization Source – A channel per source or grouped source (MPEG).



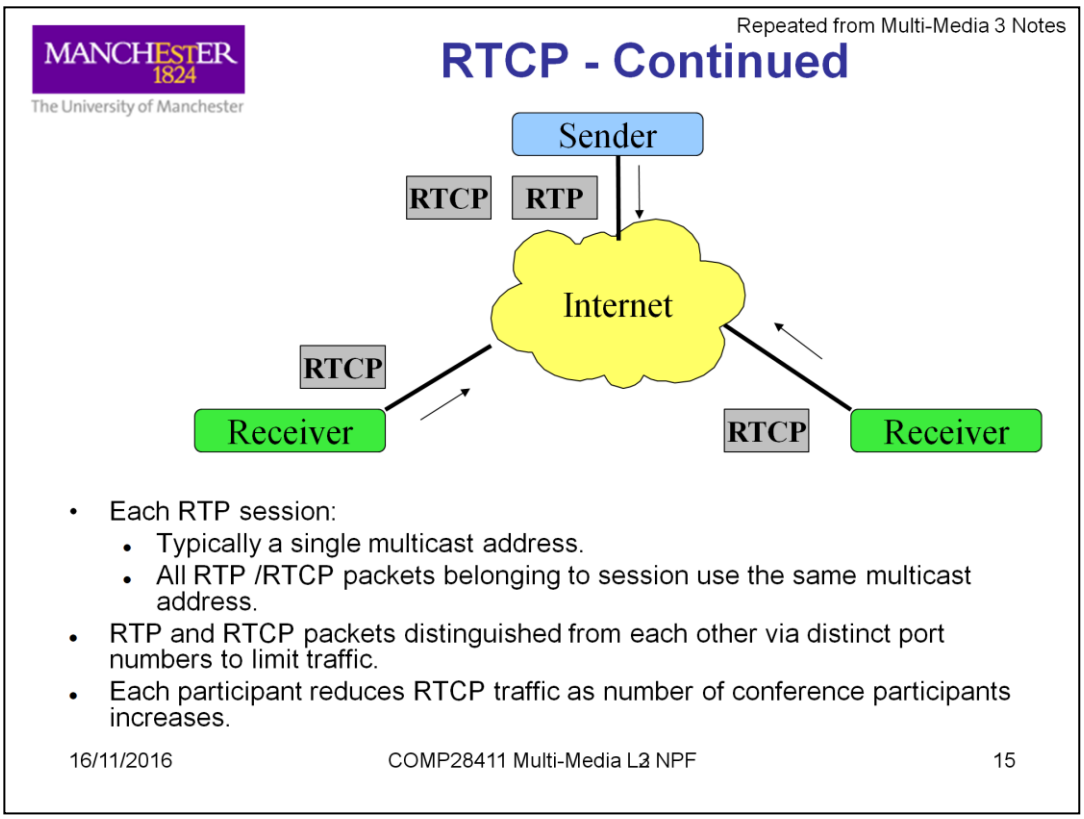


As we have already seen, RTP carries a minimal amount of information just sufficient to allow received data to be played back in the correct order and with the correct time distribution. You might query why there is both sequence number and a timestamp as the sequence number is redundant information. Why is the time stamp NOT redundant but the sequence number is?

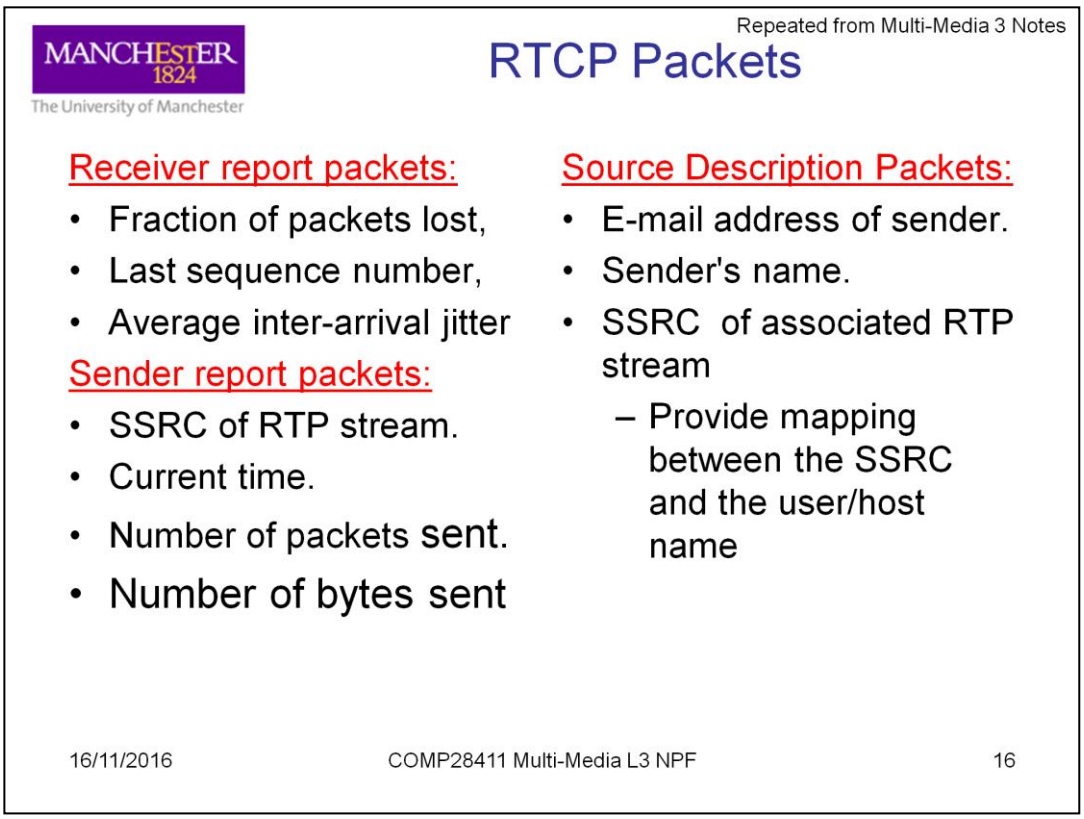
RTP is used to distribute media in many different ways. It might be unicast from 1 source to 1 destination, from 1 source to many different unicast destinations or from 1 source to an unknown number of multicast destinations. Or even broadcast to everybody. It is useful for the source to know if the destinations are getting the data and whether the data is arriving on-time or late, how the inter-packet gap is varying (jitter) and how many packets never arrive (missing sequence number). In a unicast situation this data can be sent back using RTCP every so often and used to adjust the source's choice of CODEC and rate of sending packets to try and minimise congestion and maximise the quality of the media play back.

RTCP is purely for the control of the media sent using RTP. Another protocol such as RTSP the Real-time Streaming Protocol is used to carry out tape-recorder type operations such as start, stop, seek, rewind, fast-forward on the stream. RTSP commands only make sense with unicast streams and in fact most streaming uses unicast to facilitate this and also so as to allow people to start and stop watching or listening whenever they like. Multicast streams make sense for live media and for some bulk distribution of streaming data to content delivery systems nearer the locations where the data will be played back. These content delivery systems are used to reduce network load on servers and on the network interconnections via distribution.

With multicast, the number of destinations may vary with time. Therefore mechanisms to limit RTCP feedback from destination nodes are require so as not to overload the media source with RTCP traffic.



Multicast play-back is all sent to a single network address. Destinations must register their wish to receive the multicast with their local router which in turn registers the path with router between the source and the current router. Multicast allows a single packet to be sent and duplicated on need to deliver to many destinations. RTP and RTCP packets all use the same multicast IP address so they use the same routers and destinations. Using different usually adjacent ports is enough to separate the two streams of packets from each other. However, as more destinations register for the multicast, existing destinations must reduce the amount of feedback traffic they generate based on being told via RTCP the total count of destinations for the multicast.



The content of the RTCP packets is fairly obvious and easy to guess. You can look up the exact fields and formats used on the Internet. The data is almost all metrics or statistics on the media transfer.

Repeated from Multi-Media 3 Notes

MANCHESTER 1824
The University of Manchester

Synchronization of Streams

- RTCP can synchronize different media streams within an RTP session
- Consider a video-conferencing application for which each sender generates separate RTP streams for video and for audio.
- Timestamps in RTP packets tied to the video or audio sampling clocks.
  - **Not** tied to wall-clock time
- Each RTCP sender-report packet contains (for most recently generated packet in associated RTP stream):
  - Time-stamp of RTP packet
  - Wall-clock time for when packet was created.
- Receivers use association to synchronize play-out of audio and video.

16/11/2016 COMP28411 Multi-Media L3 NPF 17

By using the same multicast address as the RTP uses for RTCP packets, these packets go to everybody in the session. This is great for conferencing as sender reports contain sequence number, timestamp and wall-clock time associated with the last packet sent. This then allows fairly exact synchronization of media coming from many different sources to be synchronized at destinations so that fully interactive sharing of what appears to be a single media stream can be achieved. The various streams ae mixed together for play back each obeying its own time stamp constraints with the local destination buffering able to adjust the different streams to align them at least roughly in time. Clearly this is not sufficient for high fidelity mixing of e.g. a distributed classical music orchestra or choir. Such an application would need much tighter constraints so that all instruments are heard to start at the same time. However, they still do not need to start exactly in synchronization as our ears are used to some sound distribution in large concert halls of many milliseconds (sound does only about 343m/s).

Repeated from Multi-Media 3 Notes

RTCP Bandwidth Scaling

- RTCP attempts to limit its traffic to 5% of session bandwidth.

Example

- Suppose one sender, sending video at 2 Mbps. Then RTCP attempts to limit its traffic to 100 Kbps.
- RTCP gives 75% of rate to receivers; remaining 25% to sender
- The 75 kbps is equally shared among receivers:
  - With R receivers, each receiver gets to send RTCP traffic at 75/R kbps.
- Sender gets to send RTCP traffic at 25 kbps.
- Participant determines RTCP packet transmission period by calculating the average RTCP packet size (across entire session) and dividing by allocated rate.

16/11/2016 COMP28411 Multi-Media L3 NPF 18

This is a simple example of how RTCP controls its amount of traffic in a distributed environment. The only issue left is how the value R is found. I think it is simple enough – and there are hints in my earlier notes. Think about who hears each multicast transmission.

MANCHESTER 1824
The University of Manchester

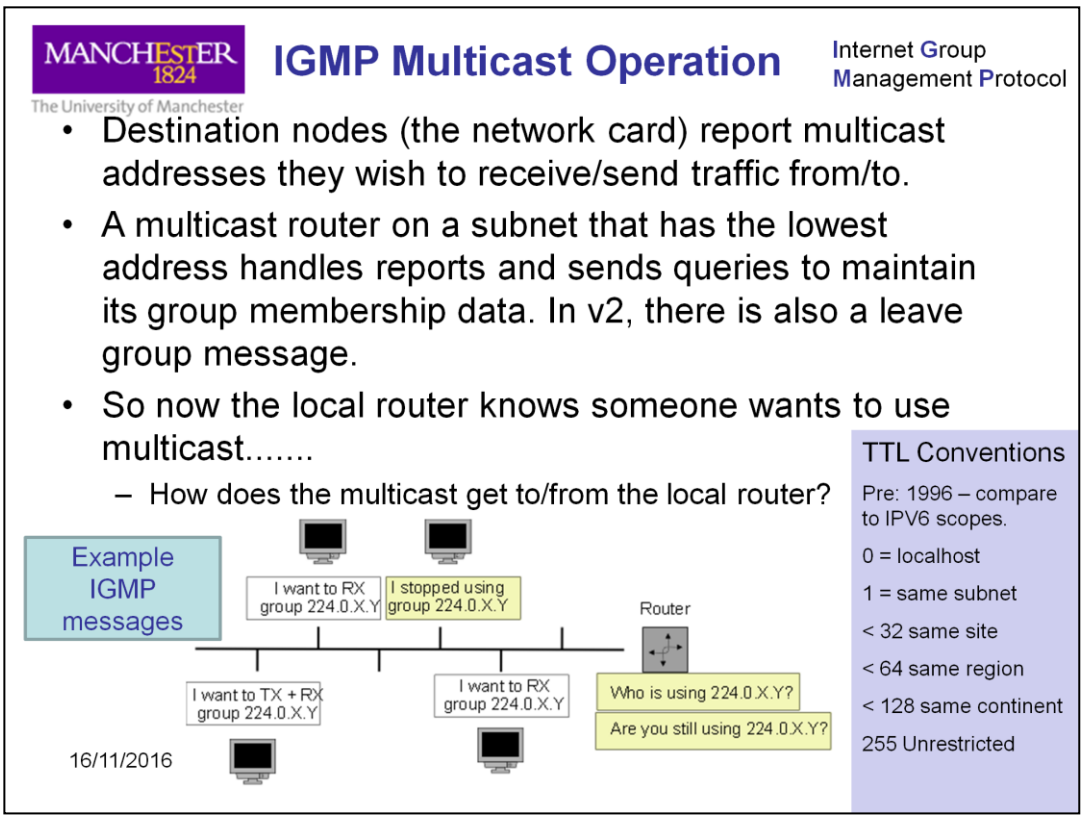
Problem ????

MULTI-MEDIA VIA UDP MULTICAST

Real Time Messaging Protocol (RTMP)

- Developed by Macromedia for server to player streaming in “Flash”. See: http://tinyurl.com/j2tptw3
- Versions for insecure, secure, over HTTP …..
- Splits media into fragments sent over separate TCP connections.
  - Default 64 bytes audio or 128 bytes video
  - Fragments interleaved (spread) and then multiplexed into separate streams.
  - Aims to meet bandwidth, latency and other constraints using these multiple fragments sent in different streams to the player.
  - Gives strong and adaptable reaction to congestion and its removal across the set of TCP connections.
  - Can of course, change codecs to drop/increase the quality as the channels behaviours change.

This is now a very popular streaming protocol since it was made partially public. Suspect it or similar protocols are behind most fast media streaming that is done today by Amazon, Netflix, TV channels ….



TTL ? Time to live.

In a LAN multicast can be a very useful way to distribute information to a varying set of listeners. The Internet Group Management Protocol manages multicast on LANs running in the LAN's router which connect it to the outside world but also typically handles lots of local system management tasks such as DHCP and DNS.

On hosts be they senders or receivers for multicast they register with the local router that they either send multicast, wish to receive it or can do both for a given multicast address. The router, after the senders and receivers have registered their interest in a multicast address can answer simple questions about the usage of the multicast address.

On the LAN, multicast packets are most probably broadcast to everybody. But intelligent switches may be able to filter (not forward) these broadcasts if no one connected downstream of the switch has registered an interest in the multicast address. Network cards (NICs) where the physical and data-link layers are normally implemented can choose to either forward arriving frames to the higher layers in the network stack or to ignore them. Some network connections can also operate in "promiscuous" mode whereby even frames which the device, in theory, has no interest in because they are not for it, can be kept and forwarded right up to applications. Wireshark (https://www.wireshark.org/), the packet sniffer tool we used in laboratory 1 uses this mode of operation when allowed to monitor all the traffic on a given network connection.

MANCHESTER 1824
The University of Manchester

Routing Between Sender and Receivers - 1

- Sender – Anywhere on the Internet
- Receivers – Can be anywhere
  - Both sender and receivers register with local routers using IGMP
  - How do they find one-another?
- Simplest and least efficient is FLOODING:
  - From source node(s) tell all neighbours, forward to all there neighbours until all connected nodes in tree/graph have received the search discarding repetitions).
    - Simple – no routing table needed – just record of recently seen addresses – but even this may be large!
    - Scaling issue – lots of duplication.

Example: Digital Living Network Alliance (DLNA) for sharing of digital media between devices <phone, computer, TV, Camera> uses multicast and Universal Plug and Play (UPnP) for discovery + control.

16/11/2016 COMP28411 Multi-Media L2 NPF 22

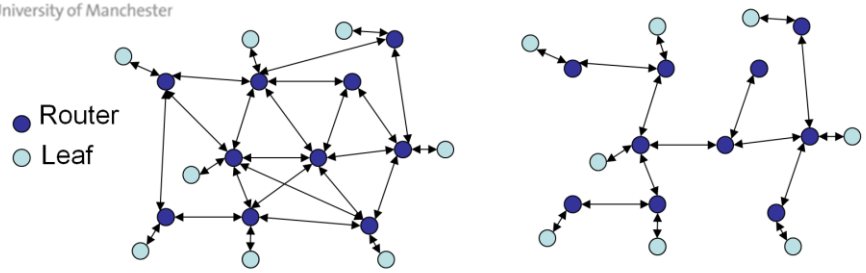
Outside of a LAN, multicast senders and receivers could be anywhere in their LAN environments accessing the outside world via the local router. The simplest search mechanism for LAN's' to find multicast users is to try everybody. For a LAN this can be reasonably efficient but between LANs on the Internet it would be very very inefficient to have to flood search packets looking for multicast users. But this method is simple and it takes quite a lot of thoughts and effort to do better. In practice, it seems that there is no one method that is always best. Therefore we will very briefly look at a range of ideas for finding multicast members and trying to find efficient routes with which to interconnect them.

The problem is made much worse because multicast streams do not have to be setup and left static throughout there operation. Multicast users can join and leave at any point. They can also be mobile, forcing the routes to/from the multicast users to vary over time. This makes delivering multicast across the Internet very difficult and unable to avoid having to search. Whenever search is necessary, it is impossible to guarantee to complete the search within a pre-determined latency/delay.

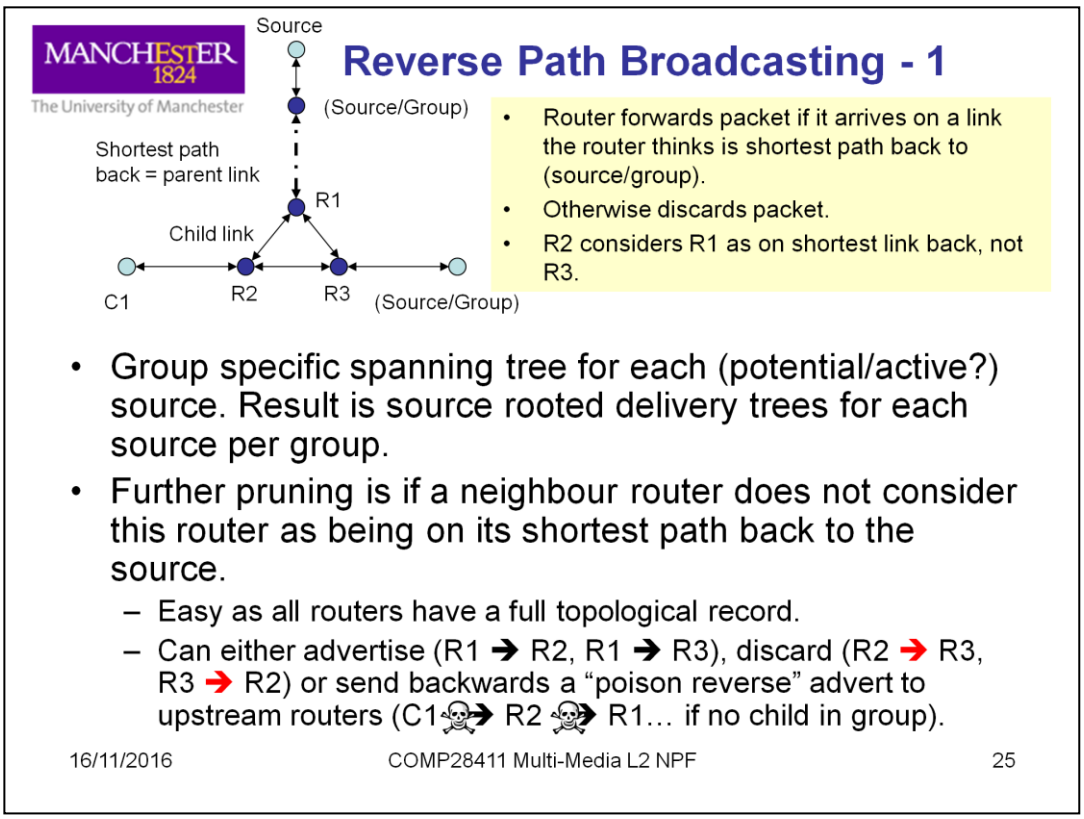
MANCHESTER 1824
The University of Manchester

Spanning Tree

P



- Select a sub-set of connected nodes – spanning tree.
- Multicast router forwards packets to all nodes that are part of the spanning tree (not the one it arrived on!).
  - No loops
  - Still reaches all the tree eventually.
  - Lots of good algorithms to find (*minimum*) spanning trees.
  - But, concentrates traffic on a few core routes, not always the best path from source to all members.



C1 is not using this multipath source!

Spanning trees provide a good step towards a solution but there can be many possible such trees with different roots. Ideally, the root of the tree should be the source or one of the sources of multicast packets. We can take this idea further by treating each multicast source in a group as the root of its own spanning tree. We end up with a set of trees each idealized for one multicast source.

The method on this slide is for reverse routing from the receivers/clients of the multicast group back to a source for the multicast group. Routers on the path will receive packets for forwarding. In this case, each router chooses to only forward packets that arrive on a route that the router believes is the shortest path back to the router. Other packets from the multicast source arriving at the router are simple discarded. The method quickly filters the used routes in the graph to those giving the best routes after first expanding to reach every node in the tree.

The expansion for our graph went:

Step1: group_member_top → R1, Step 2: R1 → R2 and R2 → R3, Step 3: R2 → C1, R3 → R2 (already visited), R3 → group_member_right.

Step 1: source_right → R3, Step 2: R3 → R1, R3 → R2, Step3: R1 → group_member_top eventually, R1 → R2 (already visited) , R2 → C1.

We assume the algorithm discards new search expansions to already visited devices.

C1 is not a group member. So it sends back a poison/prune message which traverses discarding multicast forwarding entries at routers until it reaches either a router with a child somewhere who is a member of the group or the original source node for the expansion. The decision of when to discard the multicasts at a route and prune is dealt with in a few slides time properly.

Routers cannot discard forwarding instructions until they know definitively that they have no children leading to group members.

Reverse Path Broadcasting - 2

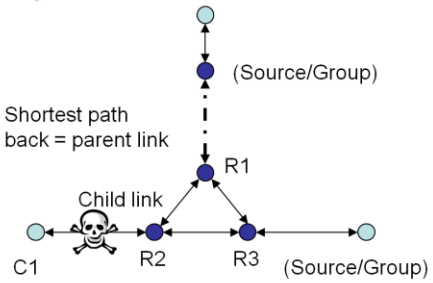
- Reasonably efficient, quite simple.
- Routers only need relatively local knowledge.
- Packets always follow “shortest” (?) path.
- Packets from different sources in same group follow different trees/paths so lower likelihood of bottleneck.
- However, the forward adverts still happen whether or not the sub-tree has a group member.
  - R2 & C1 are searched before pruning later.

16/11/2016 COMP28411 Multi-Media L2 NPF 26

This seems quite a good method. But it is not without efficiency issues. Also, it works well when setting up a multicast distribution. It will respond sensibly to nodes leaving the multicast. But, when nodes join they must initiate a search outwards to find a router that is forwarding the multicast to other places in order to join. In the worst case, this can search the whole network.

When is the worst case? I think it is when the 1st non source node joins. At all other times there are some routers with knowledge of the multicast somewhere in the network graph?

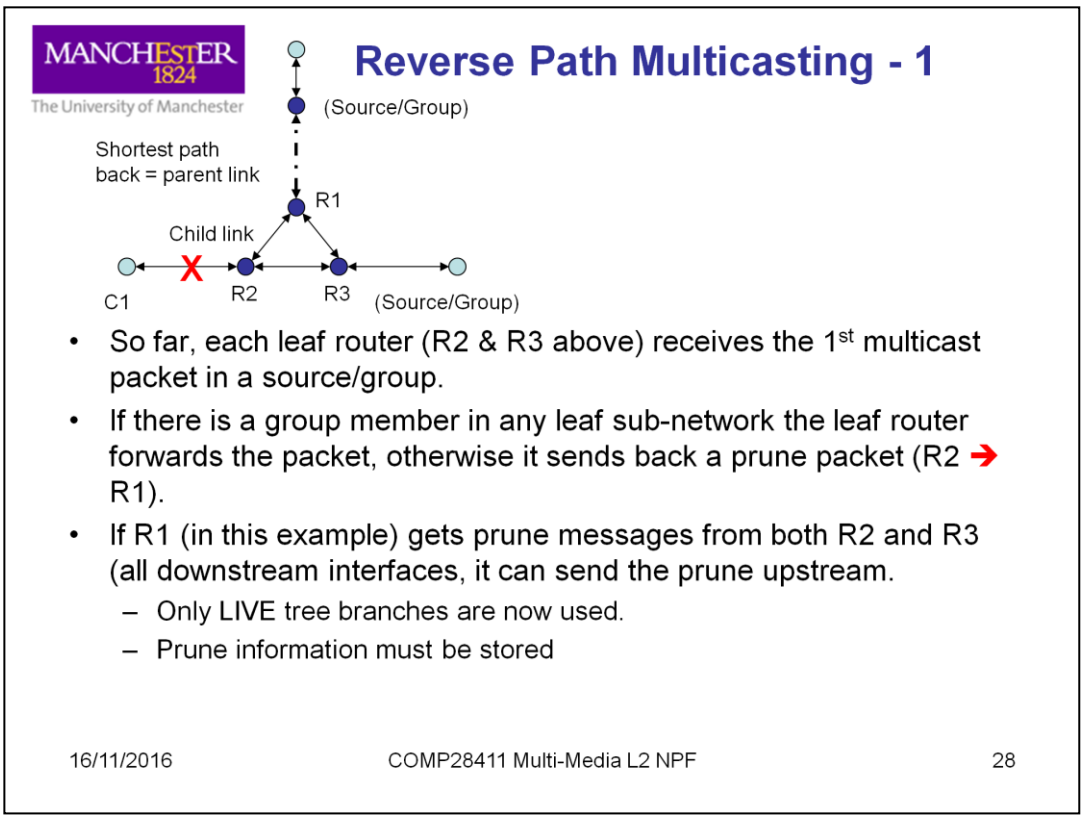
Truncated Reverse Path Broadcasting



- Using IGMP, multicast routers work out the group memberships for leaf networks. So avoid forwarding if no current members of the group.
- The sub-network tree is pruned by the router.
- But only the final router does this removing some traffic on leaf sub-networks. (R2 does this)

16/11/2016 COMP28411 Multi-Media L2 NPF 27

When IGMP is used, we could assume that R2 already knows that C1 and no other nodes in its sub-tree are interested in this multicast group. Therefore, the search can be truncated before it reaches to absolute leaf nodes of the tree. This saves quite a lot of work as these nodes are typically very numerous. But it does not save very much time as these nodes are the final hop in each direction.



This shows the correct rule for a router to discard multicasts for a particular source rooted spanning tree. Routers can prune, if all there downstream branches have no interest in the multicast packets. Otherwise, the router must forward to any router that does have multicast members somewhere downstream from it.

Reverse Path Multicasting - 2

- Because both the tree and the source + group memberships change the multicast tree must be **refreshed regularly**. This results in a full flood and prune.
- There are still scaling issues in the flooding and pruning.
- All routers need to keep information for each group and all its sources.
- In practice may need to cache/group before flooding of add & prune messages to reduce control traffic but at the expense of some extra latency.

16/11/2016 COMP28411 Multi-Media L2 NPF 29

Regular tree refreshing is needed. This can be initiated by a timer or by listening for join/leave requests. Both methods mentioned involve some extra latency which seems to be unavoidable unless we are happy to do a complete network search for each group membership change!

This is perhaps why, in practice, multicast is almost never used on the complete unstructured Internet! Instead we use it frequently now for LAN systems and organizations use it across highly filtered wide area distribution networks using virtual overlay networks or a technology similar to an overlay network the Virtual Private Network (VPN). A VPN, effectively tunnels packets through large jumps across the Internet thus vastly reducing the number of nodes that need to be searched for multicast membership.

Core Based Routing

- A set of static “centres” are identified at different locations in the network. Multicast traffic is duplicated and may be unicast to these centres.
- Branches grow from core routers towards multicast users. Traffic is multicast from core routers to the leaves of the tree.
- This is more scalable.
  - Router only keep group, not source per group, information.
  - Full tree flooding is avoided as “centre” are statically defined and can be unicast to. Multicast is only used in local clusters.
- Issues:
  - Traffic concentration near core routers. Traffic from all sources uses common links towards core routers.
  - Shared links can create sub-optimal routes and therefore delays.

16/11/2016 COMP28411 Multi-Media L2 NPF 30

Using the idea form the previous slide gives further ideas for controlling the multicast group searches. A set of pre-chosen nodes is identified as good places to send the traffic to be multicast to before starting to multicast it. Distribution networks often provide a set of well connected servers near core populations. By sending the traffic to these distribution centres the size and range of the multicast searches can be massively reduced and limited in depth; they only need to be as deep as the biggest gap between adjacent distribution centres.

However, this does have a down side in that it will tend to concentrate traffic towards and from these distribution centres. The distribution centres need to have high capacity Internet links to lots of routers.

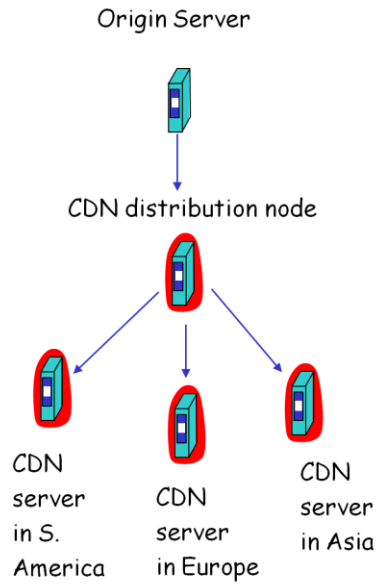
MANCHESTER 1824
The University of Manchester

Problem ????

HOW IS MASS DEMAND CONTENT DISTRIBUTED TO MILLIONS OF USERS

Content distribution networks (CDNs)

- Content replication
  - Challenging to stream large files (e.g., video) from single origin server in real time
  - *Solution:* replicate content at hundreds of servers throughout Internet
    - Content downloaded to CDN servers ahead of time
    - *Placing content "close" to user avoids impairments (loss, delay) of sending content over long paths*
    - CDN server typically in edge/access network



16/11/2016 COMP28411 Multi-Media L2 NPF 32

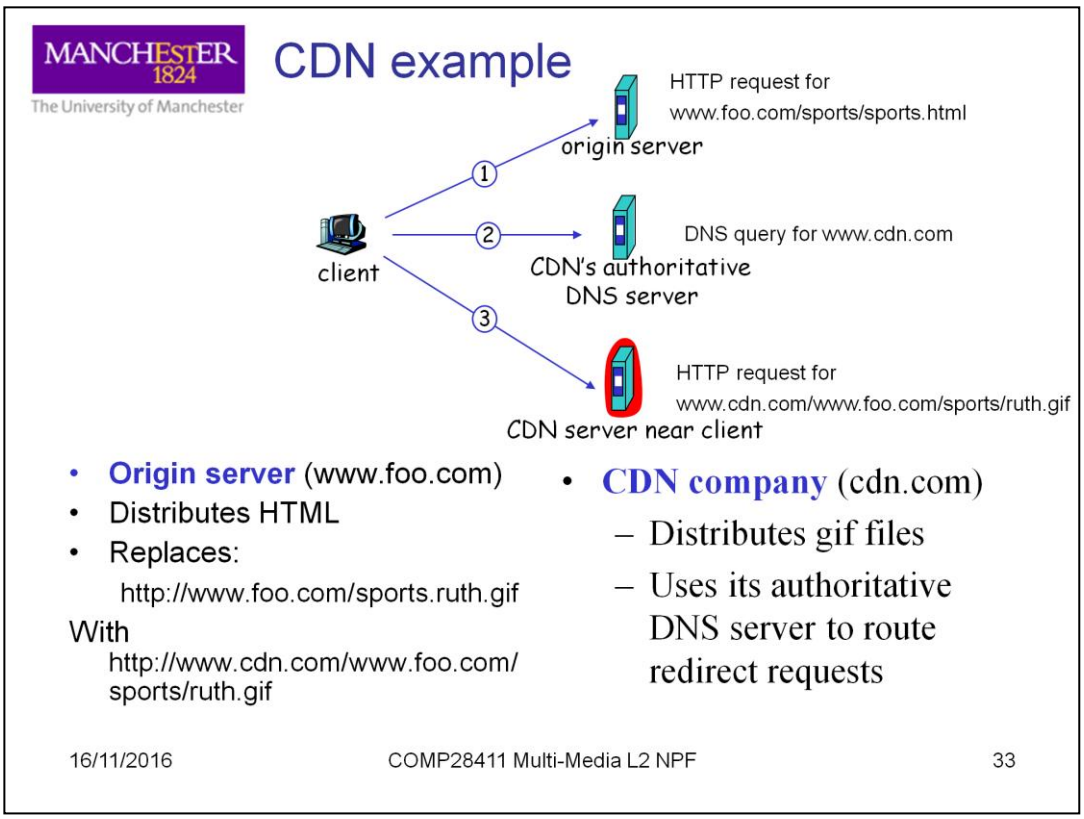
We have talked about this several times during the recent lectures on Multi Media. Here we briefly illustrate the solution that is used.

Not compulsory but well worth a look:

Open Connect Everywhere: A Glimpse at the Internet Ecosystem through the Lens of the Netflix CDN 2016

Timm Böttger, Felix Cuadrado, Gareth Tyson, Ignacio Castro, Steve Uhlig

http://tinyurl.com/jbcu8lm



Most CDN's use DNS to rewrite queries from the wanted web site to redirect to their web sites.

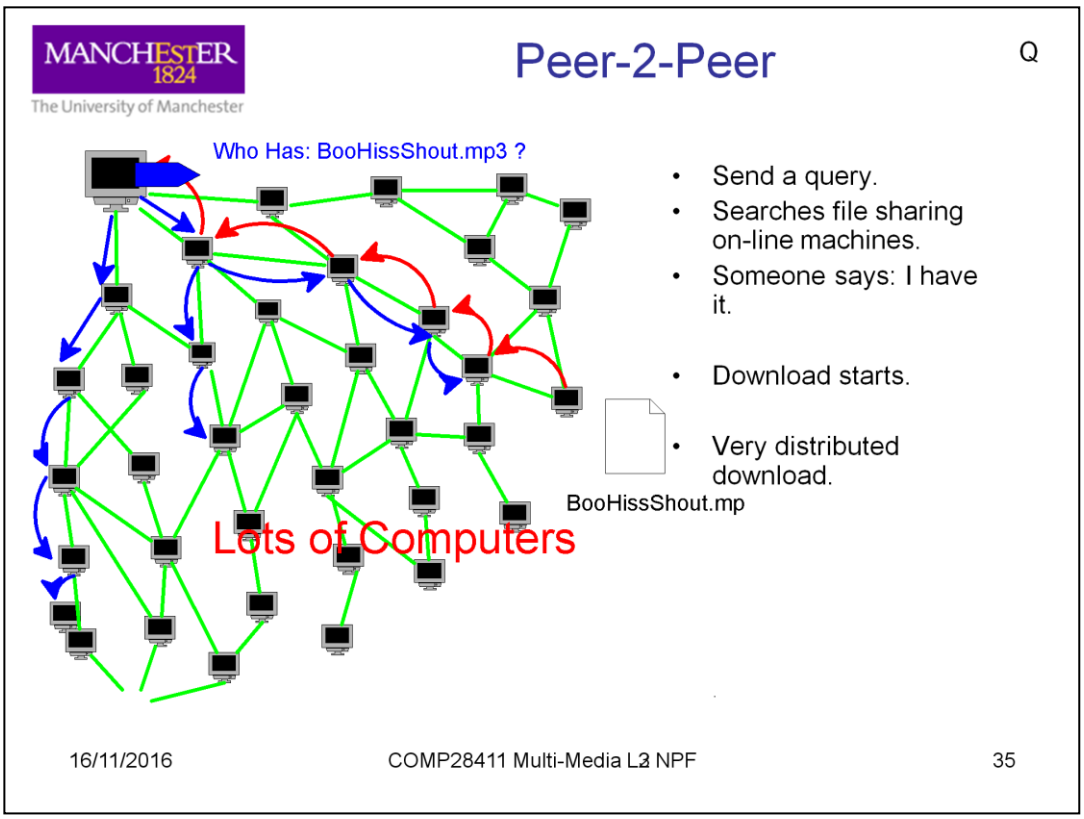
The authoritative DNS server for the original web site is probably an Anycast address which redirects the traffic to a nearby server owned or run by the CDN provider. At the redirect, the URL can also be renamed allowing different CDN companies to take queries in different locations. Some large Internet companies are now also CDNs. I believe this is true for Google and Amazon. See: https://en.wikipedia.org/wiki/Content_delivery_network @ http://tinyurl.com/7oc4pg8

MANCHESTER 1824
The University of Manchester

More about CDNs

- **Routing requests**
  - CDN creates a “map”, indicating distances from leaf ISPs and CDN nodes
  - When query arrives at authoritative DNS server:
    - Server determines ISP from which query originates
    - Uses “map” to determine best CDN server
  - CDN nodes create application-layer overlay network

Overlay networks are very common on today's Internet. The overlay imposes a different addressing scheme onto a subset of the real underlay Internet's hosts. This simplifies data handling for large systems. It provides routes between distributed hosts which are to a greater or lesser extent independent of IP addresses. Many of these systems use a HASH method that uses meta-data for the real data such as the filename, its type and its content to generate an address in the overlay network. By including the IP address mapping between the underlay Internet and the overlay network into the address generation process it is possible to push data towards locations in the underlay network. This mechanism is often used to push the storage location for data nearer to its users.

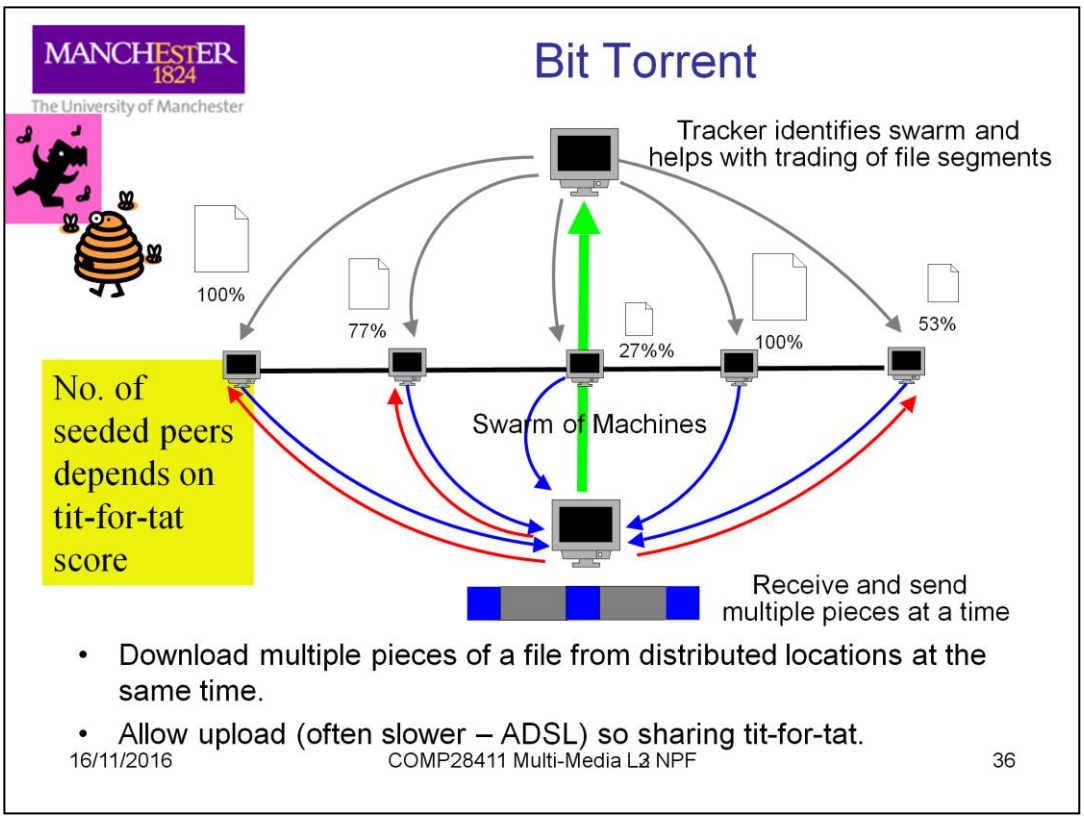


We call these overlay networks Peer to Peer (P2P) networks. There are many different examples of these systems. Early P2P systems were often unstructured which means that whilst the data is distributed there is no good algorithm to determine where the data is stored from e.g. its name. This means a random search is needed or maybe a graph flash search. These are of course quite inefficient.

Other P2P systems (most modern ones) are based on various data structure topologies such as rings, trees, threaded trees, meshes and graphs. Being structured means that knowing what is wanted, an algorithm can direct a search to the data in typically O(n) when the amount of data is small enough to have a single table lookup mechanism based on hashing again or O(log n) when only part of the address space can be searched at each step.

One example is a system called CHORD. CHORD has a massive 2^{160} address space which is organized into a ring. Hosts are distributed around the ring based typically on hashing their IP address into the ring space. Files are assigned to hosts for storage by hashing their name, and contents to generate addresses. Each node has a pointer table that maps CHORD addresses (160 bit values) to IP addresses of hosts. The table is typically quite small but maps to the actual address where the data is stored in O(log N) time where N is the total number of hosts.

Systems like CHORD make storage and retrieval of massive numbers of data files relatively efficient when it would be impossible to store all the files at a single location. Used for music, video and similar data.



The CHORD P2P system is fully distributed in that there is no central server. Bit torrent is related to this but does have a central server though it used to do only a tiny part of the retrieval and storage tasks. The idea for bit torrent is to store files in many locations, for example, with hosts that use the files on a regular basis. Then when someone wants to download a copy of the file, instead of doing so from a single server, parts of the file are downloaded from a large set of different servers distributed across the Internet. This has a big advantage for the Internet load because it distributes the traffic such that the different parts of the file being downloaded come from different locations over different paths. This maximizes the parallel load balancing and also can maximise the download speed. However, for us at home, the bottleneck remains as the link from our home computer to the telephone exchange.

Bit torrent varies the use of copies of the wanted file using a tit-for-tat system. The more you let others upload the data from your hosts, the more different fragments from different locations are requested. This means people who let others upload from their hosts get much higher download speeds. This encourages fair sharing.

MANCHESTER 1824
The University of Manchester

Summary

- Delivering media is a continually developing solution.
- Demand, location, time of day, density of users all effect choice of method.
- We have:
  - TCP – End-to-end unicast, reliable but subject to delays, duplication, bottlenecks (hot-spots).
  - UDP - End to end unicast, unreliable, less delay problems due to accepting losses. Avoid duplication partially via multicast.
  - By sending multiple TCP streams side by side we can increase throughput or add redundancy to improve some Quality of Service (QoS) except in really bad congestion.
    - **Making it work needs fragmentation of the input media streams.**
    - **Plus then needs to be interleaved across multiple streams so if one or more streams losses data (congestion) others will hopefully succeed. The application still gets some data even if not all of it.**

MANCHESTER 1824
The University of Manchester

Questions? Things to thinks about.

- How would you change the equations for “adaptive play-out delay” to adapt rapidly to delay variations?
- When are multi-media applications likely to use TCP in preference to UDP?
- Is RTP ever used or useful embedded in TCP packets?
- An audio steam samples at 4000Hz. Suggest a suitable timestamp clock increment gap for RTP to use for this audio stream?
- Often RTP SSRC values are chosen randomly. How might a receiver detect and resolve SSRC value collisions?

MANCHESTER 1824
The University of Manchester

More Questions?

- QoS needs packets to be marked (class of service). Where are the markers put?
- What does policing do when it detects a stream using too many resources?
- Why is priority based scheduling sometimes unfair?
- A Weighted Fair Queue (WFQ) ensures the outbound link to a network is not over used. What happens if too much inbound traffic arrives?
- How does a token bucket aid in ensuring fair usage even when streams exceed there normal arrival rate?
- Explain why token bucket and WFQ together can guarantee an upper bound on the delay to packets passing through a router?

MANCHESTER 1824
The University of Manchester

Yet More Questions?

- Most media is stored in the cloud. Why is this often a good idea currently? When is it not that good an idea?
- Why do you think a large amount of jitter is bad for media players?
- Why doesn't uTube use mulitcast?
- Why is multicast little used currently? (not easy to find much on this)
- How does UPnP work? Why is it suited for media handling tasks?
- Is tele-conferencing currently mainly unicast or multicast, P2P or centralized?