The Transport and Application Layers

COMP 30650: Networks and Internet Systems

Dr. Gavin McArdle
Email: gavin.mcardle@ucd.ie
Office: A1.09 Computer Science

RECAP

Transport Layer

- End-to-End Communication
- Ports
- Protocols
  - UDP
  - TCP
  - Reliabilty
- Improving Performance
  - Sliding Window

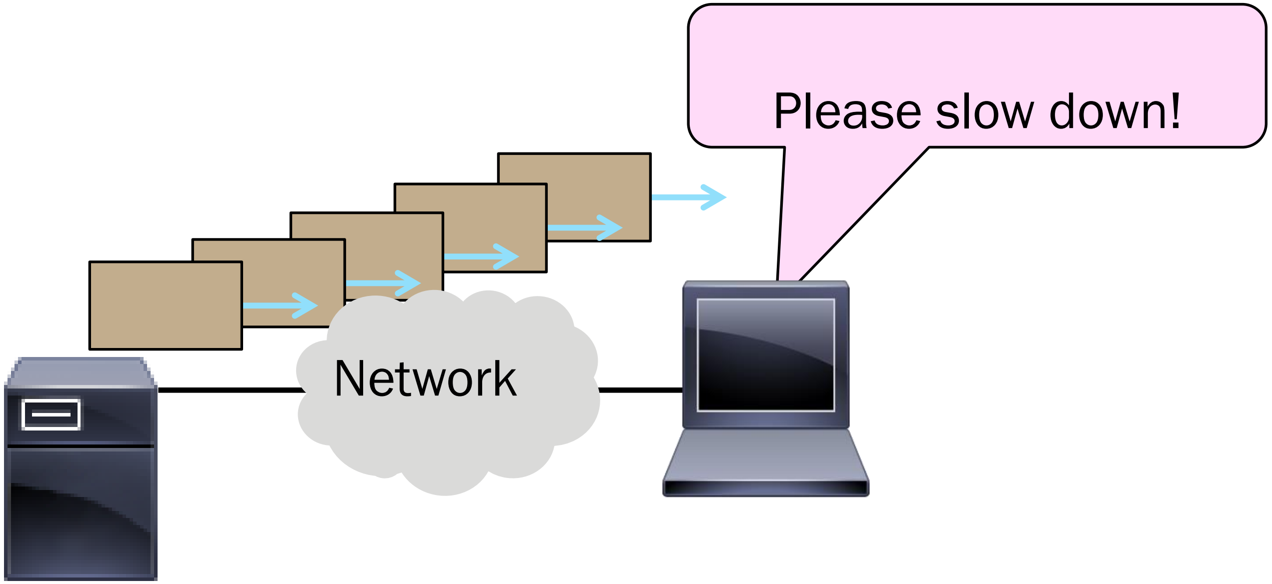
TODAY'S PLAN

- Improving Performance
  - Flow Control
- Application Layer
  - DNS

FLOW CONTROL

Adding flow control to the sliding window algorithm

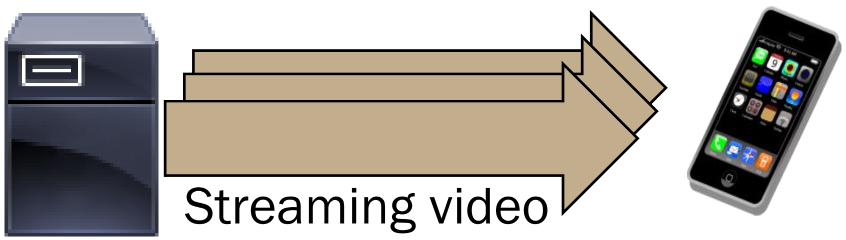
- To slow the over-enthusiastic sender



PROBLEM

Sliding window uses pipelining to keep the network busy

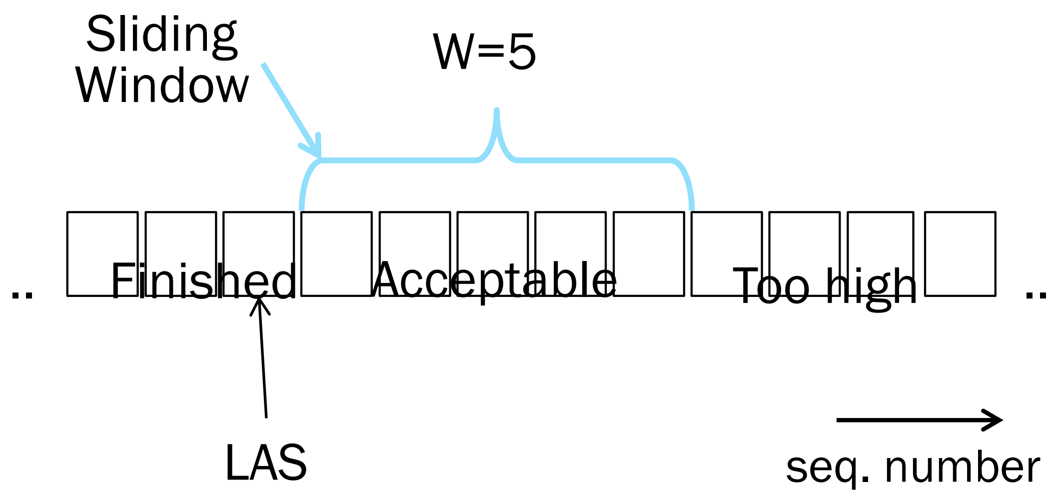
- What if the receiver is overloaded?



SLIDING WINDOW – RECEIVER

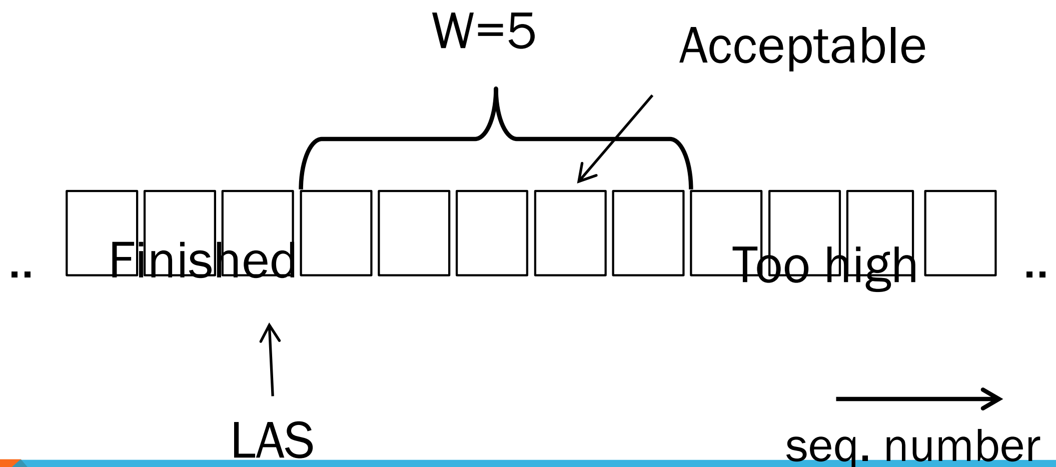
Consider receiver with W buffers

- LAS=LAST ACK SENT, app pulls in-order data from buffer with recv() call



SLIDING WINDOW – RECEIVER

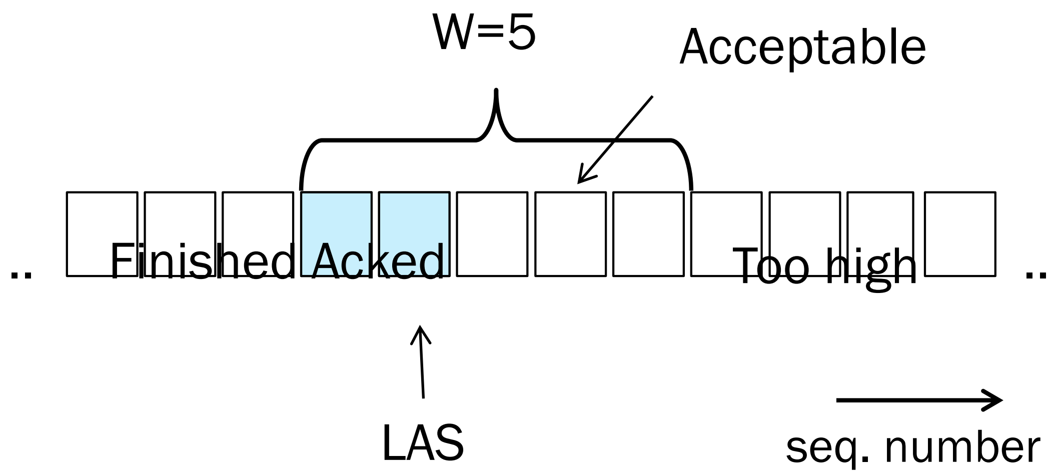
Suppose the next two segments arrive but app does not call recv()



SLIDING WINDOW – RECEIVER

Suppose the next two segments arrive but app does not call recv()

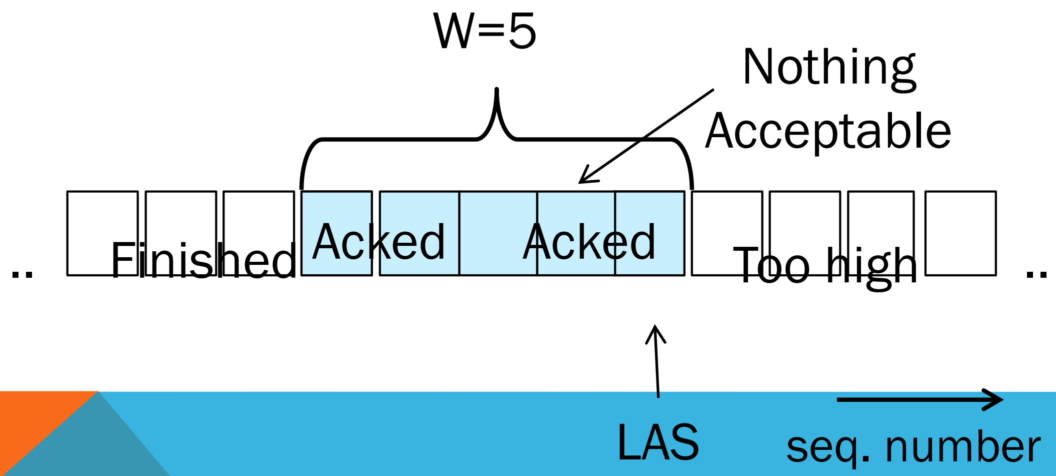
- LAS rises, but we can't slide window!



SLIDING WINDOW – RECEIVER

If further segments arrive (even in order) we can fill the buffer

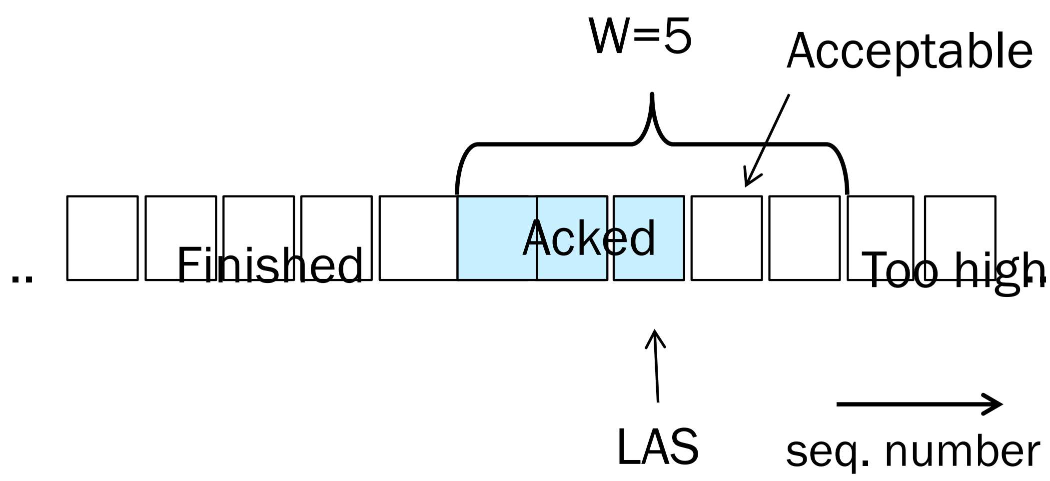
- Must drop segments until app recvs!



SLIDING WINDOW – RECEIVER

App recv() takes two segments

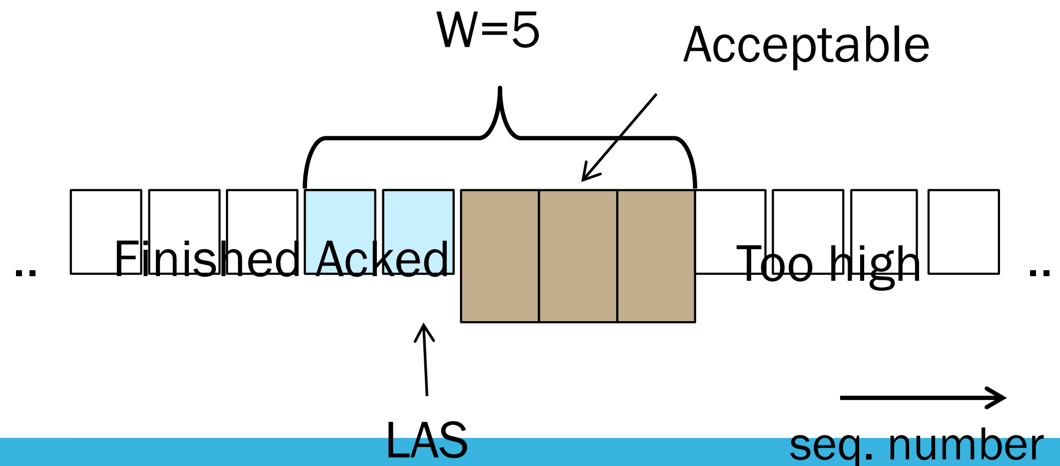
- Window slides



FLOW CONTROL

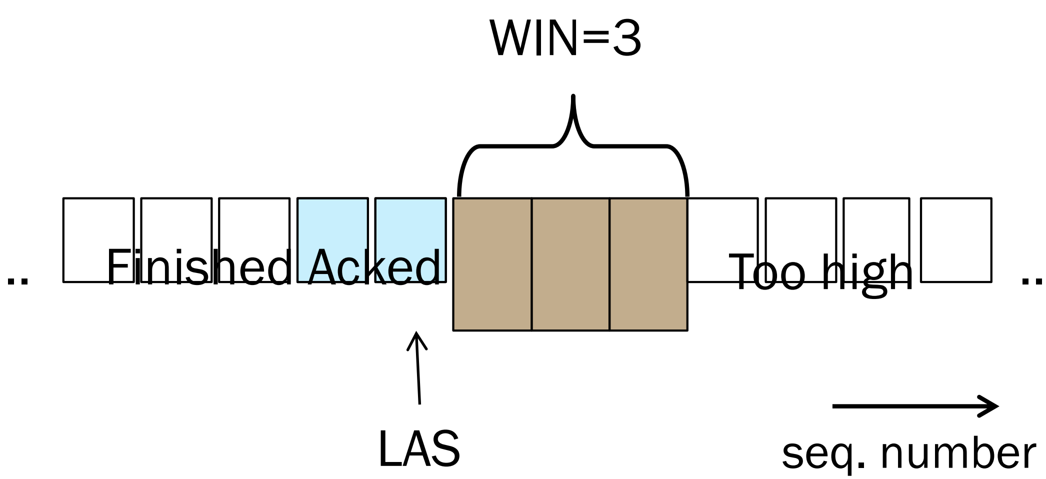
Avoid loss at receiver by telling sender the available buffer space

- WIN=#Acceptable, not W (from LAS)



FLOW CONTROL

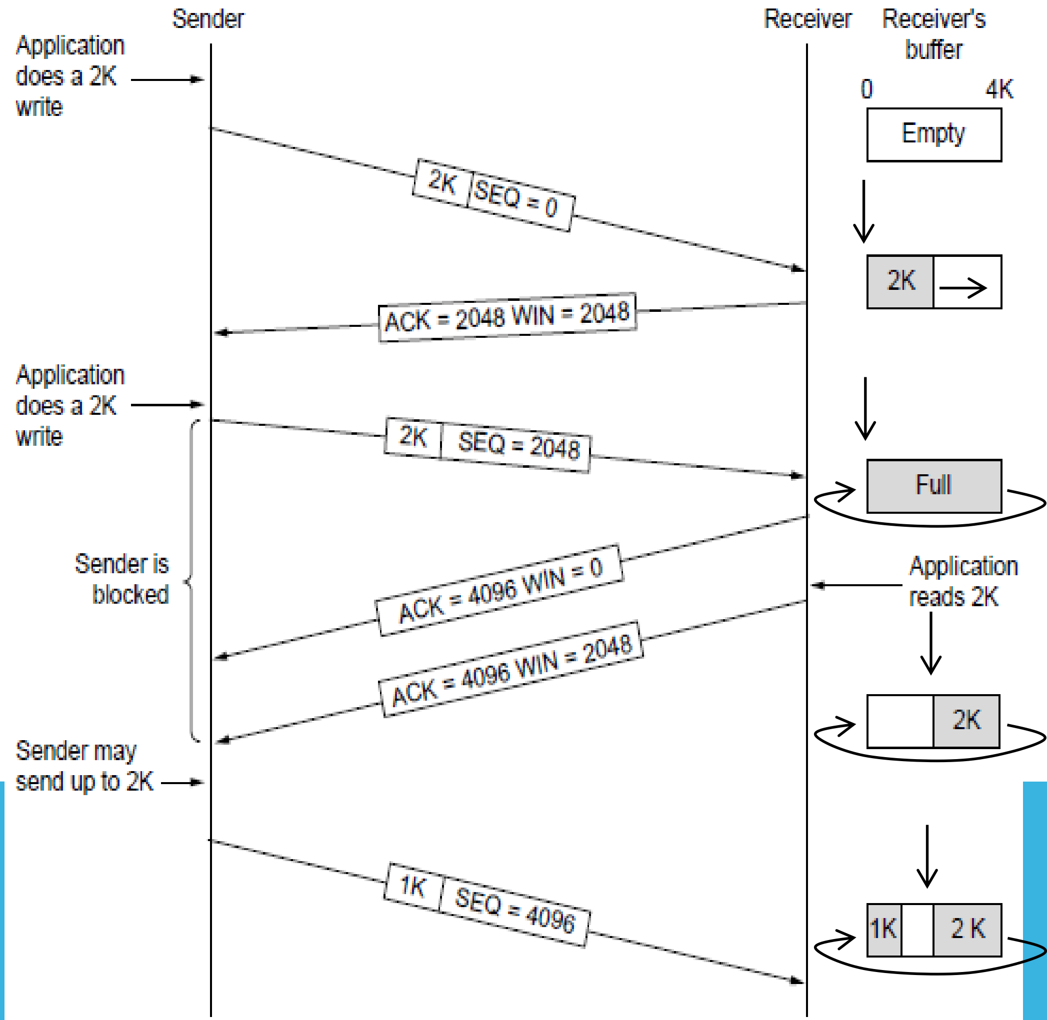
Sender uses the lower of the sliding window and flow control window (WIN) as the effective window size



FLOW CONTROL

TCP-style example

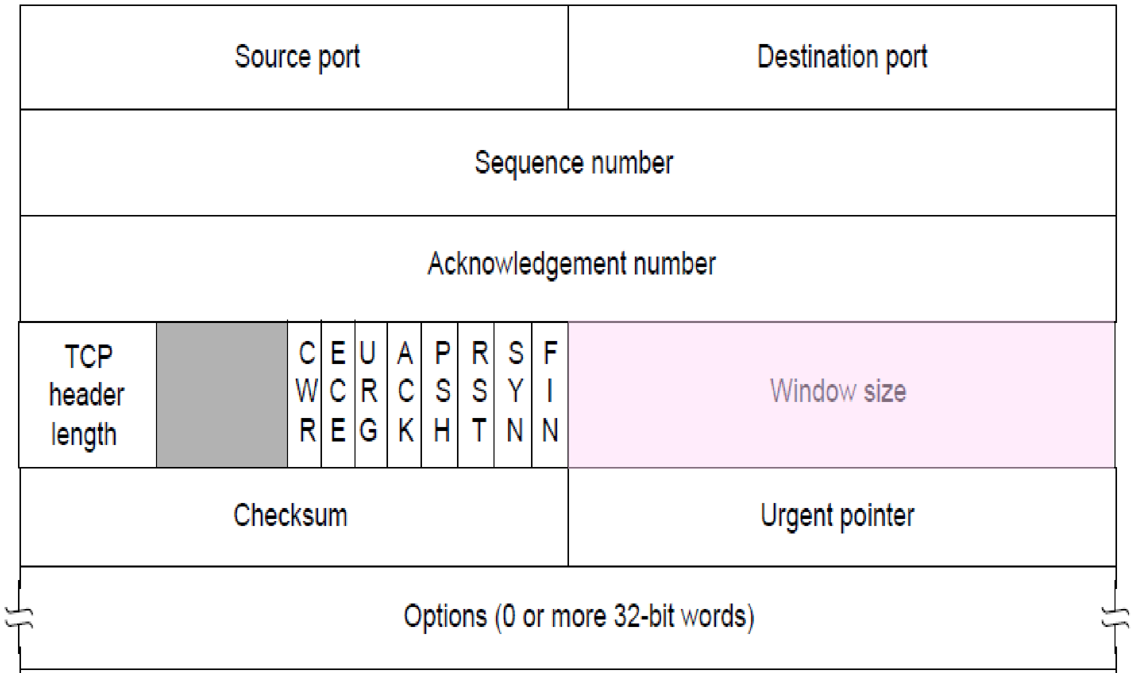
- SEQ/ACK sliding window
- Flow control with WIN
- SEQ + length < ACK+WIN
- 4KB buffer at receiver
- Circular buffer of bytes



TCP HEADER

Window size for flow control

- Relative to ACK, and in bytes



The Application Layer

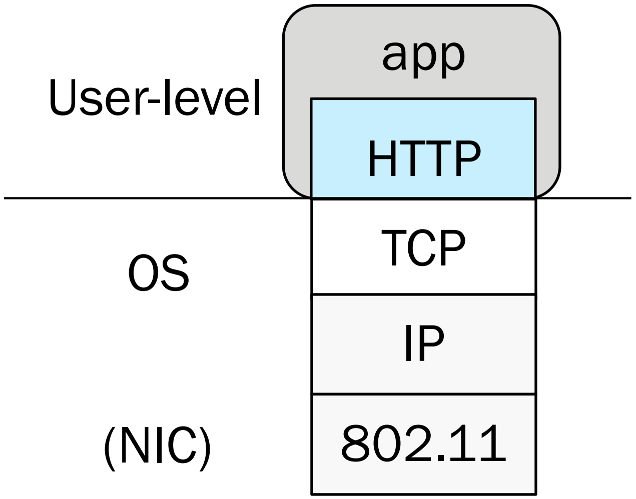
COMP 30650: Networks and Internet Systems

Dr. Gavin McArdle
Email: gavin.mcardle@ucd.ie
Office: A1.09 Computer Science

RECALL

Application layer protocols are often part of an "app"

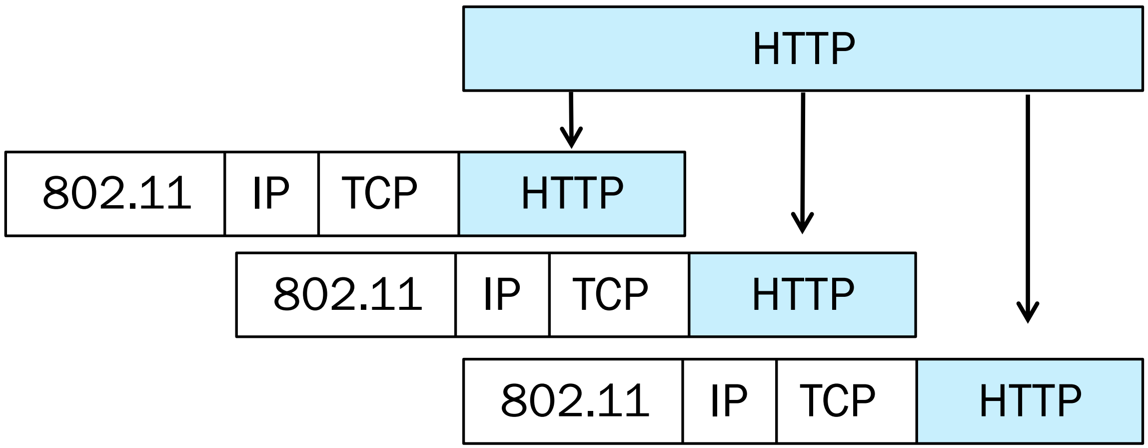
- But don't need a GUI
  - DNS
- Some have GUIs
  - Web browser



RECALL

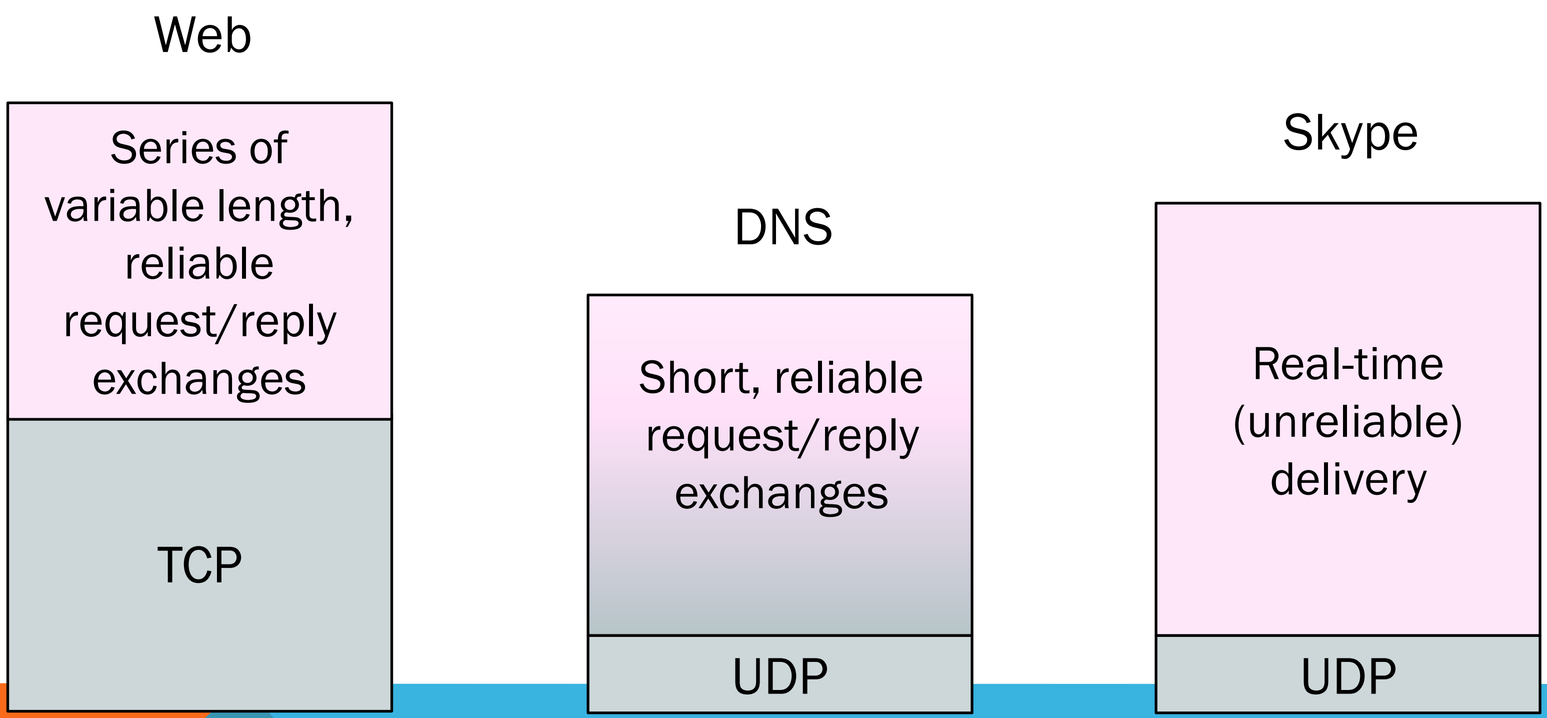
Application layer messages are often split over multiple packets

- Or may be aggregated in a packet …



APPLICATION COMMUNICATION NEEDS

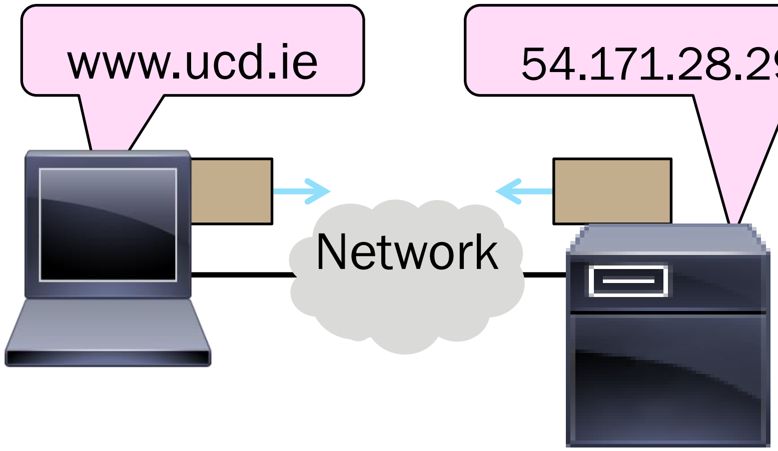
Vary widely with app; must build on Transport services



FILLING IN THE GAPS OF OUR BIG PICTURE

The DNS (Domain Name System)

- Human-readable host names



NAMES AND ADDRESSES

Names are higher-level identifiers for resources

Addresses are lower-level locators for resources

- Multiple levels, e.g. full name → email → IP address → Ethernet address

Resolution is mapping a name to an address

- Lookup in an address book or directory or some sort of Table!

BEFORE THE DNS – HOSTS.TXT

- Directory was a file HOSTS.TXT regularly retrieved for all hosts from a central machine at the NIC (Network Information Center)
- Names were initially flat, became hierarchical (e.g., lcs.mit.edu)
- Neither manageable nor efficient as the ARPANET grew …

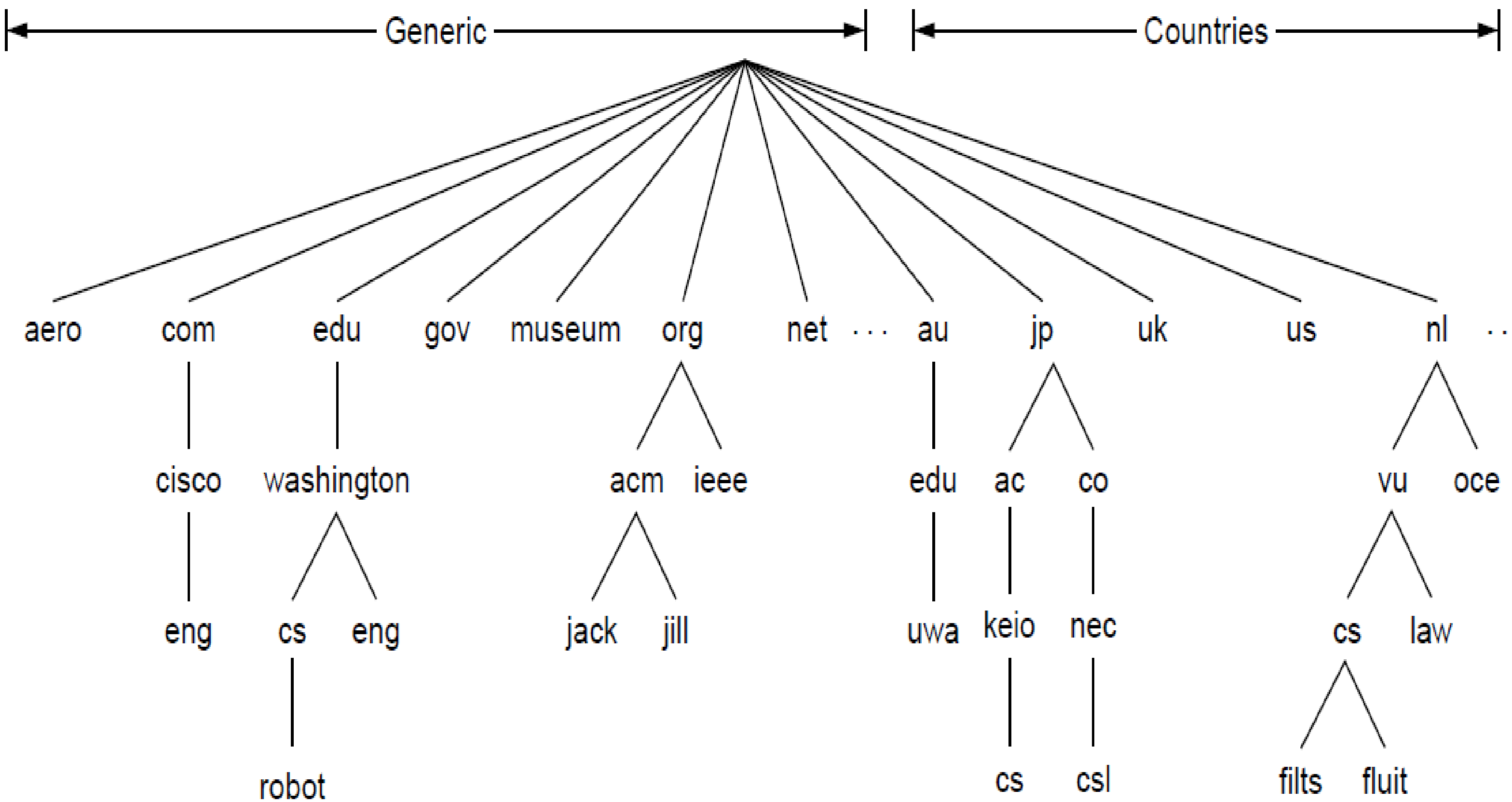
DNS

- A naming service to map between host names and their IP addresses
  - www.ucd.ie → 54.171.28.29
- Goals:
- Easy to manage (esp. with multiple parties)
- Efficient (good performance, few resources)

- Approach:
- Distributed directory based on a hierarchical namespace
- Automated protocol to tie pieces together

DNS NAMESPACE

Hierarchical, starting from "." (dot, typically omitted)



TLDS (TOP-LEVEL DOMAINS)

Run by ICANN (Internet Corp. for Assigned Names and Numbers)

22+ generic Top-Level Domains

- Initially .com, .edu , .gov., .mil, .org, .net

~250 country code TLDs

- Two letters, e.g., ".au", plus international characters since 2010
- Widely commercialized, e.g., .tv (Tuvalu)
- Many domain hacks, e.g., instagr.am (Armenia), goo.gl (Greenland)

Other domains

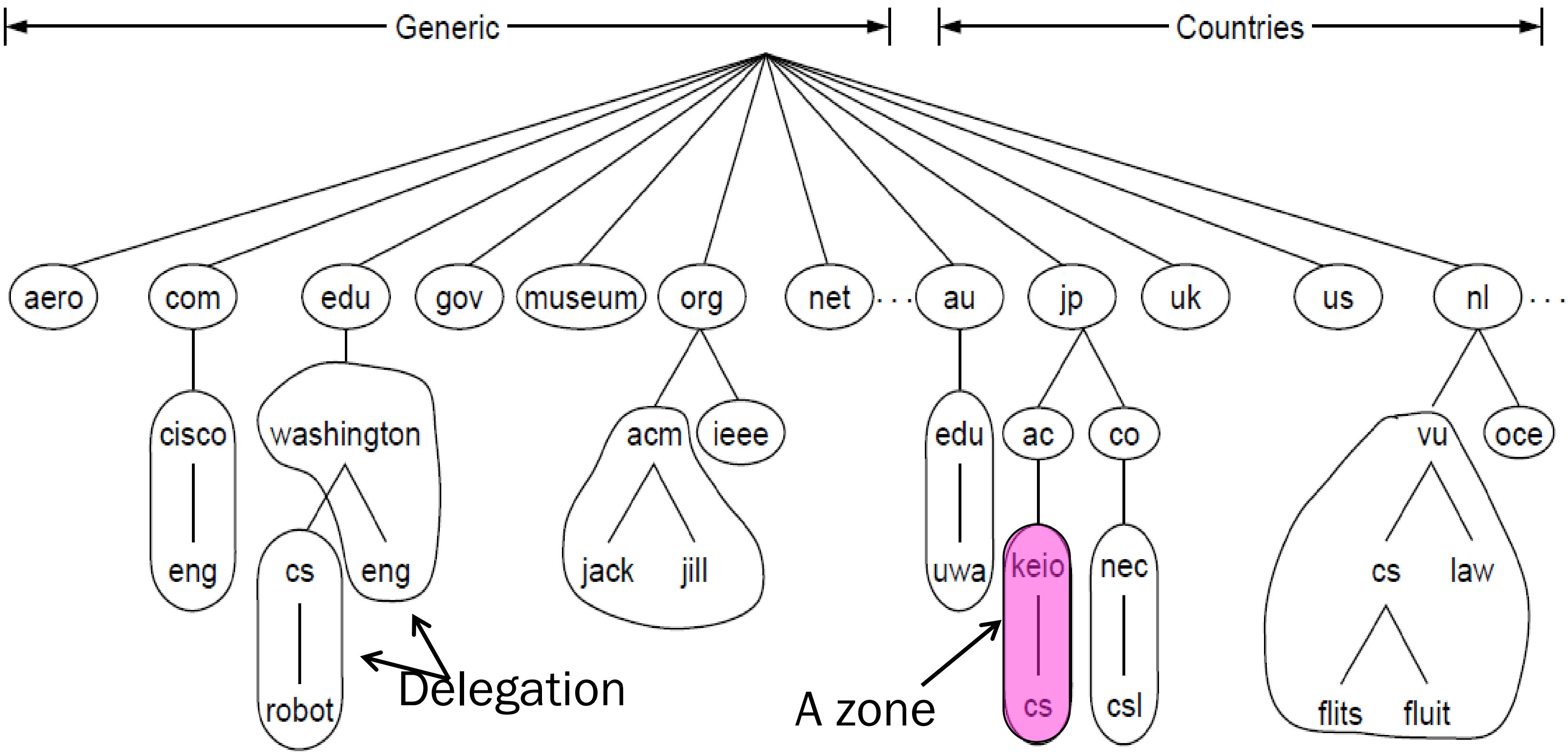
  - .academy to .zone and everything between.

Brand domains

  - google, amex, omega, etc.

DNS ZONES

A <u>zone</u> is a contiguous portion of the namespace



DNS ZONES

Zones are the basis for distribution

- EDU Registrar administers ***.edu***
- UW administers ***washington.edu***
- CS administers ***cs.washington.edu***
    - cs.uw.edu

Each zone has a <u>nameserver</u> to contact for information about it

- Zone must include contacts for delegations, e.g., .edu knows nameserver for washington.edu

DNS RESOURCE RECORDS

A zone is comprised of DNS resource records that give information for its domain names

Type	Meaning
SOA	Start of authority, has key zone parameters
A	IPv4 address of a host
AAAA (“quad A”)	IPv6 address of a host
CNAME	Canonical name for an alias
MX	Mail exchanger for the domain
NS	Nameserver of domain or delegated subdomain

DNS RESOURCE RECORDS

```
; Authoritative data for cs.vu.nl
cs.vu.nl.       86400  IN  SOA    star boss (9527,7200,7200,241920,86400)
cs.vu.nl.       86400  IN  MX     1 zephyr
cs.vu.nl.       86400  IN  MX     2 top
cs.vu.nl.       86400  IN  NS     star

star            86400  IN  A      130.37.56.205
zephyr          86400  IN  A      130.37.20.10
top             86400  IN  A      130.37.20.11
www             86400  IN  CNAME  star.cs.vu.nl
ftp             86400  IN  CNAME  zephyr.cs.vu.nl

flits           86400  IN  A      130.37.16.112
flits           86400  IN  A      192.31.231.165
flits           86400  IN  MX     1 flits
flits           86400  IN  MX     2 zephyr
flits           86400  IN  MX     3 top

rowboat                IN  A      130.37.56.201
                       IN  MX     1 rowboat
                       IN  MX     2 zephyr

little-sister          IN  A      130.37.62.23

laserjet               IN  A      192.31.231.216
```

Name server

IP addresses of computers

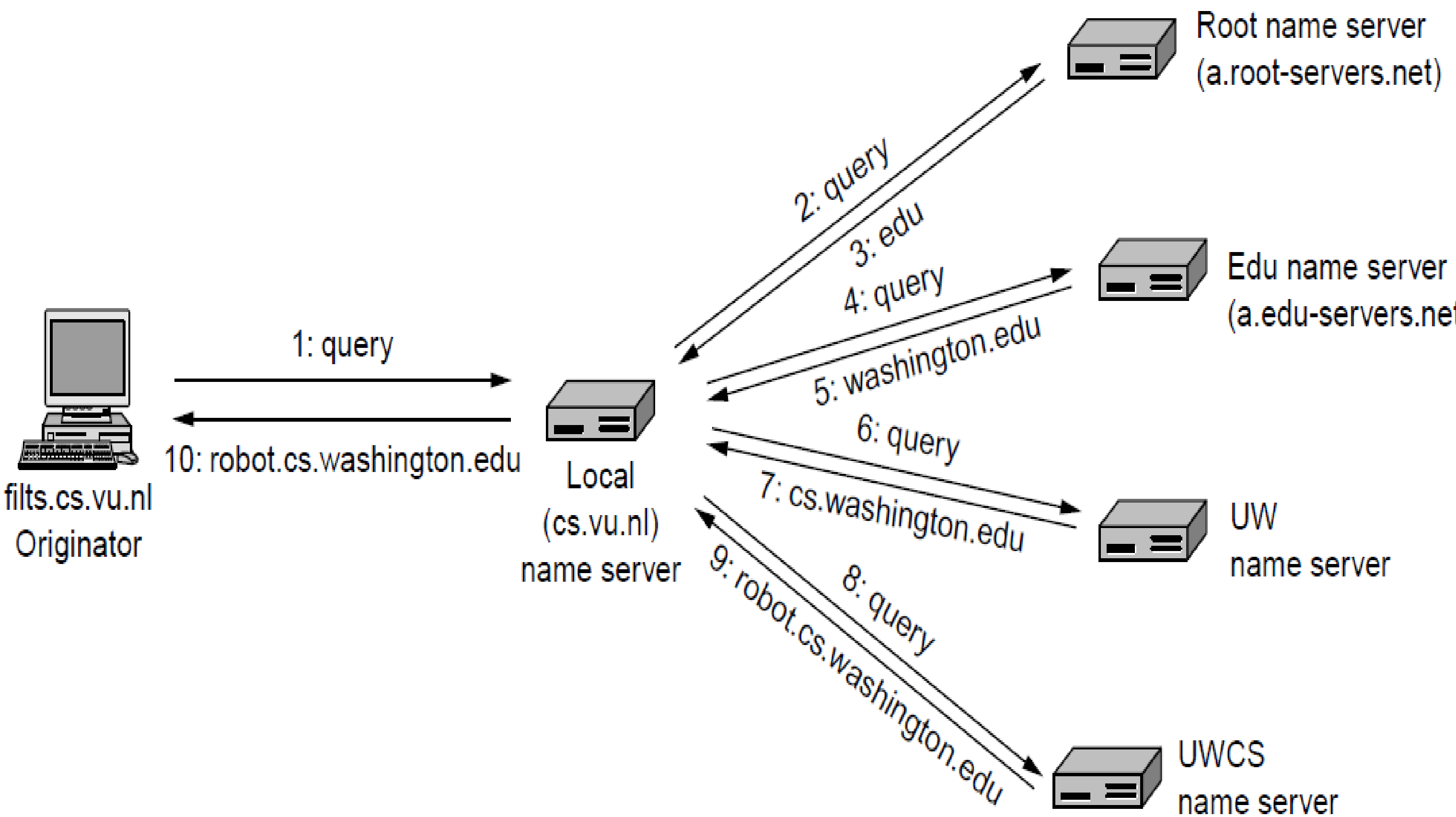
Mail gateways

DNS RESOLUTION

- DNS protocol lets a host resolve any host name (domain) to IP address
- If unknown, can start with the root nameserver and work down zones

DNS RESOLUTION

flits.cs.vu.nl resolves robot.cs.washington.edu



ITERATIVE VS. RECURSIVE DNS QUERIES

Recursive query

- Nameserver completes resolution and returns the final answer
- E.g., flits → local nameserver

Iterative query

- Nameserver returns the answer or who to contact next for the answer
- E.g., local nameserver → all others

ITERATIVE VS. RECURSIVE DNS QUERIES

Recursive query

- Lets server offload client burden (simple resolver) for manageability
- Lets server cache over a pool of clients for better performance

Iterative query

- Lets server "file and forget"
- Easy to build high load servers

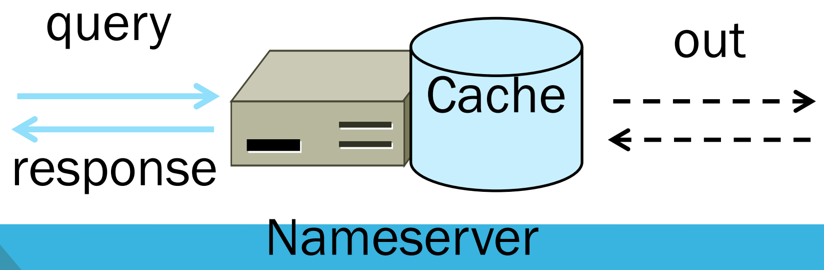
CACHING

Resolution latency should be low

- Adds delay to web browsing

Cache query/responses to answer future queries immediately

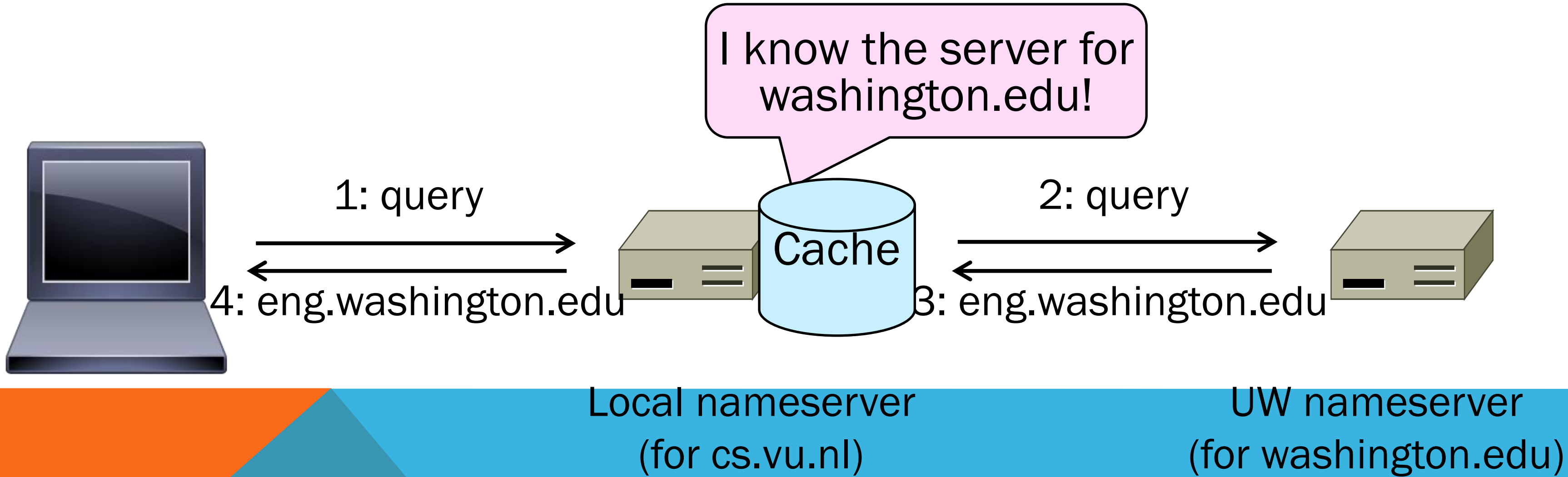
- Including partial (iterative) answers
- Responses carry a TTL for caching



CACHING

flits.cs.vu.nl now resolves eng.washington.edu

- And previous resolutions cut out most of the process



LOCAL NAMESERVERS

Local nameservers typically run by IT (enterprise, ISP)

- But may be your host or AP
- Or alternatives e.g., Google public DNS

Clients need to be able to contact their local nameservers

- Typically configured via DHCP

ROOT NAMESERVERS

Root (dot) is served by 13 server names

- a.root-servers.net to m.root-servers.net
- All nameservers need root IP addresses
- Handled via configuration file (named.ca)

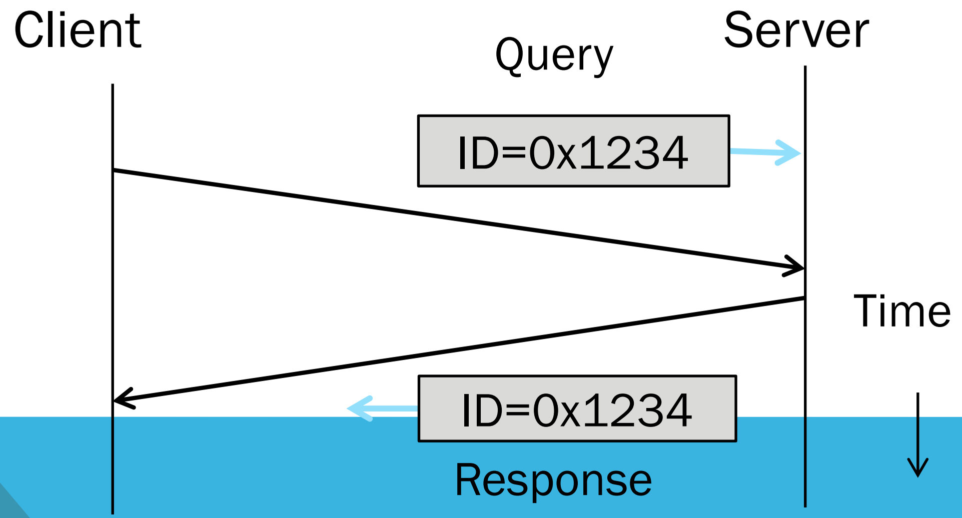
There are >250 distributed server instances

- Highly reachable, reliable service
- Most servers are reached by IP anycast (Multiple locations advertise same IP!)
- Routers take client to the closest one.
- Servers are IPv4 and IPv6 reachable

DNS PROTOCOL

Query and response messages

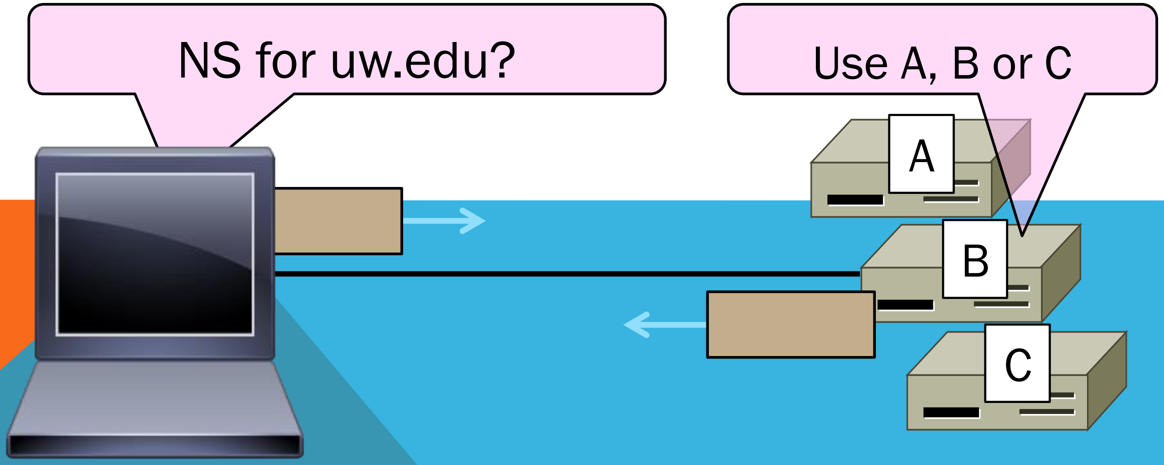
- Built on UDP messages, port 53
- ARQ for reliability; server is stateless!
- Messages linked by a 16-bit ID field



DNS PROTOCOL

Service reliability via replicas

- Run multiple nameservers for domain
- Return the list; clients use one answer
- Helps distribute load too



DNS PROTOCOL

Security is a major issue

- Compromise redirects to wrong site!
- Not part of initial protocols ..

DNSSEC (DNS Security Extensions)