Week 2

Communication and Connectivity

FIT1050 Web Fundamentals

Copyright Warning

Learning outcomes

Communication Protocols

- Understand the concept of protocols in computing
- Differentiate between different layers of protocols for Internet communication
- Understand what the purpose of the web's main protocols

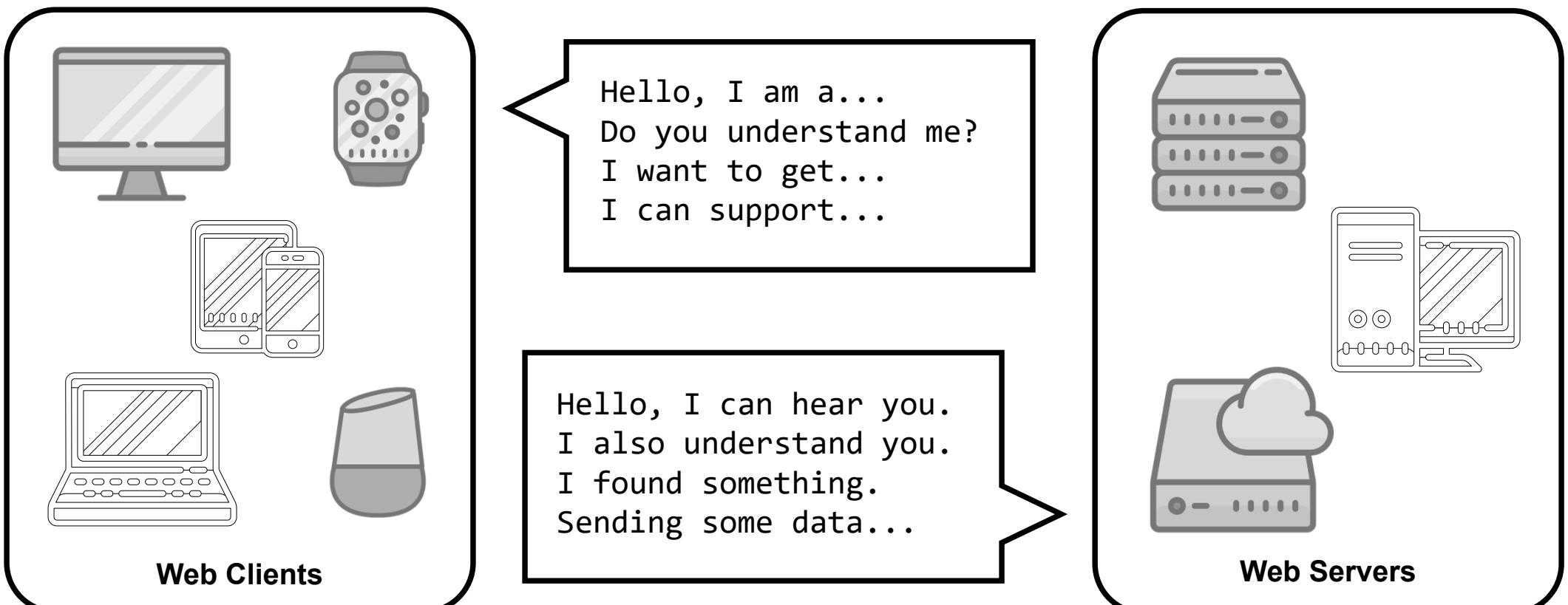
Internet Connections

- Understand the variety of connections
- Explore different issues that affect the usability of an Internet connection
- Consider how mobile connectivity affects web communication

Communication Protocols

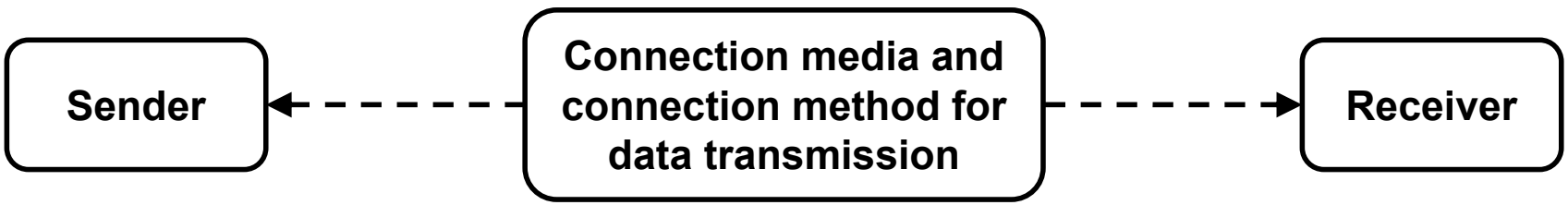
Review: clients and servers

Last week we considered how devices can communicate reliably.



The communication problem

Every information-based system has an information transfer issue:



1. What are the available methods of making a connection, so that information can be transmitted?
2. Are different methods required for different stages of the transmission?
3. What are the different strengths and weaknesses of each method?

What are the strengths and weaknesses of these?

- **Person-to-person networks**
  - Face-to-face conversation / phone call / postal service
- **Printed media networks**
  - Newspapers / magazines / catalogues
- **Electronic broadcast media**
  - TV / radio

Specific features of each method effect what can be communicated, as well as the efficiency and effectiveness of how information is sent/received.

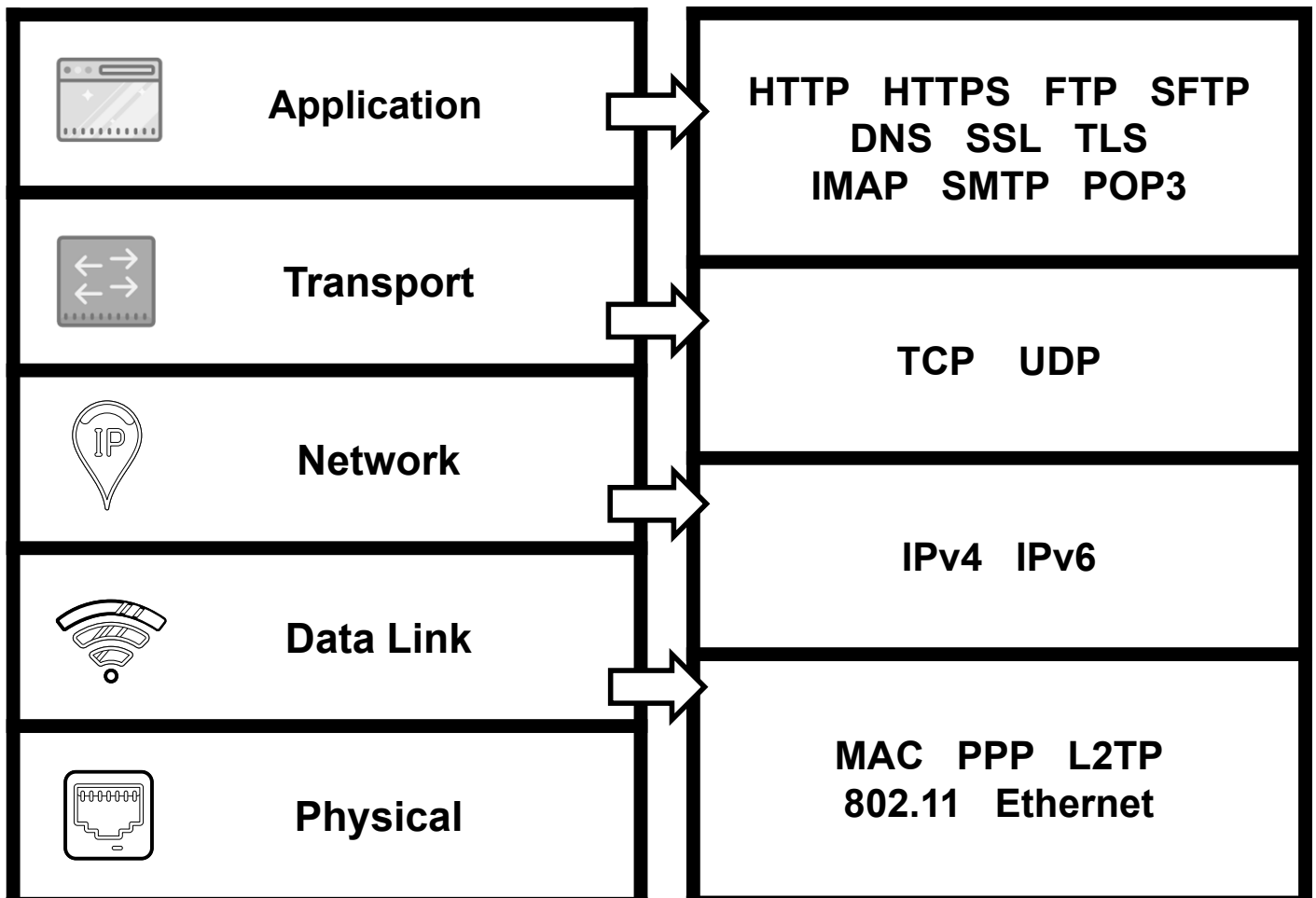
Communication standards

- **The problem of communication between separate computer networks**
  - Compatibility of communication hardware
  - Compatibility of communications methods
- **To operate many different services seamlessly…**
  - Rules to control how communication takes place
  - Clear documentation of rules
  - **Standards and protocols!**
- Protocols are task specific - designed to efficiently handle a specific single aspect of communication

Protocol layers

Instead of one complex set of rules to cover all stages of the process, we can break down the communication process into **layers**.

Then develop or choose from various technology standards to solve problems at each layer.



Some historical developments

Packet-switching (1961)

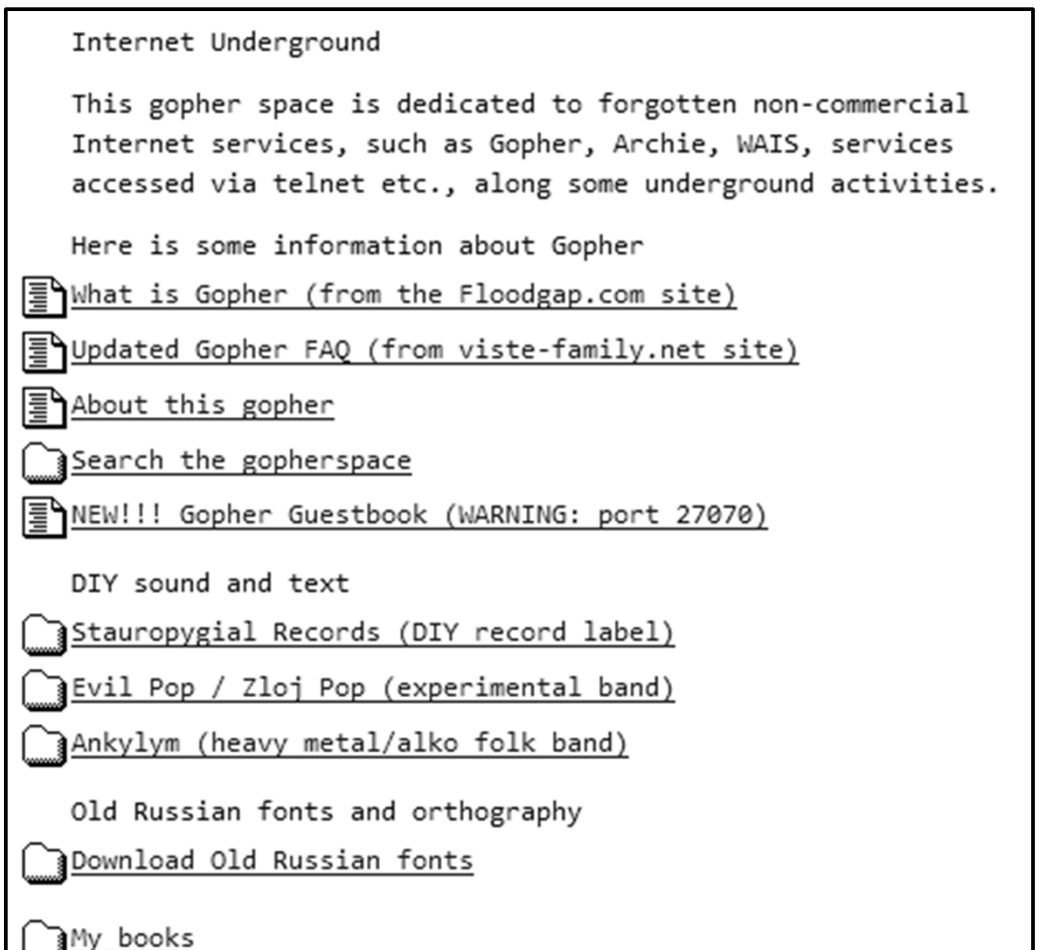
Transmitting data in small blocks (packets), developed for military use to increase network robustness

ARPANET (1969)

An early packet-switched network, and the first to implement TCP/IP. Funded by the US Department of Defence

Gopher (1991, example on right)

A protocol presented as an alternative to the World Wide Web.
Still operational, largely unchanged.



gopher.rp.spb.su via http://gopher.floodgap.com/gopher/

Network layer: Internet Protocol (IP)

This protocol serves 3 main functions:

1. **Addressing host interfaces**
    - Define a standard addressing system
    - Ensuring network hardware is uniquely addressable
2. **Fragmenting data into packets and reassembling it later**
    - To match the maximum transmission unit supported by hosts
3. **Routing data from source to destination**
    - Connections are rarely direct - often requiring many "hops".

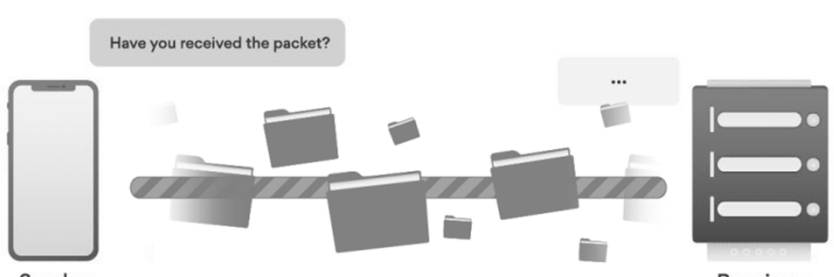
Transport layer: Transmission Control Protocol (TCP)

Manages data movement between an application and Internet Protocol while Ensures integrity for many application protocols (web, email, streaming, etc)

1. **Negotiate and establish a working connection**
   - Communicates transmission requirements between hosts
2. **Detects missing or duplicate packets**
   - Re-request missing packets, discard duplicates
3. **Detects out-of-order packets**
   - Re-sequences data packets before passing to an application.

TCP vs UDP (User Datagram Protocol)

Transport Control Protocol	User Datagram Protocol
Detects and re-requests lost packets.	Does not validate successful packet arrival.
Assigns packet sequence and corrects order.	Sends packets as a continuous stream.
Data errors handled during communication.	Data errors are handled by the application.
Requires resources to manage data transport.	Minimal resources as data moves more directly.
Used where data integrity is critical.	Used where data is needed in quickly.



https://nordvpn.com/blog/tcp-or-udp-which-is-better/

HTTP: Hypertext Transfer Protocol

HTTP provides communications rules for both requests and responses.

The client sends a request containing:

- Request metadata, including a URL
- Optional encoded data

The server sends a response including:

- Request metadata, including a status code
- The requested data, usually in the form of an HTML document

HTTP authentication

HTTP does provide basic authentication (login) features via the request URL.

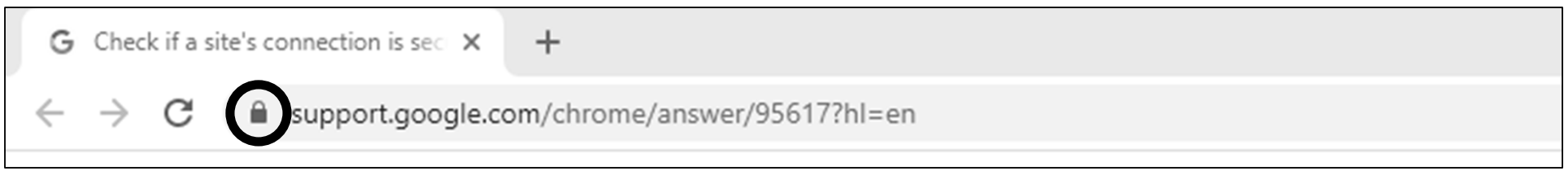
```
http://username:password@www.example.com/restricted/
```

However, this method is not recommended (at least on its own)

- Using plain HTTP the request URL is visible in transit.
- On the client, the credentials are visible in the browser's address bar.
- On the client, the credentials are stored in the browser's history.
- At the server, the credentials will be logged in plaintext.
- At the server, there is no default mechanism for preventing brute-forcing.

HTTPS - Hypertext Transfer Protocol Secure

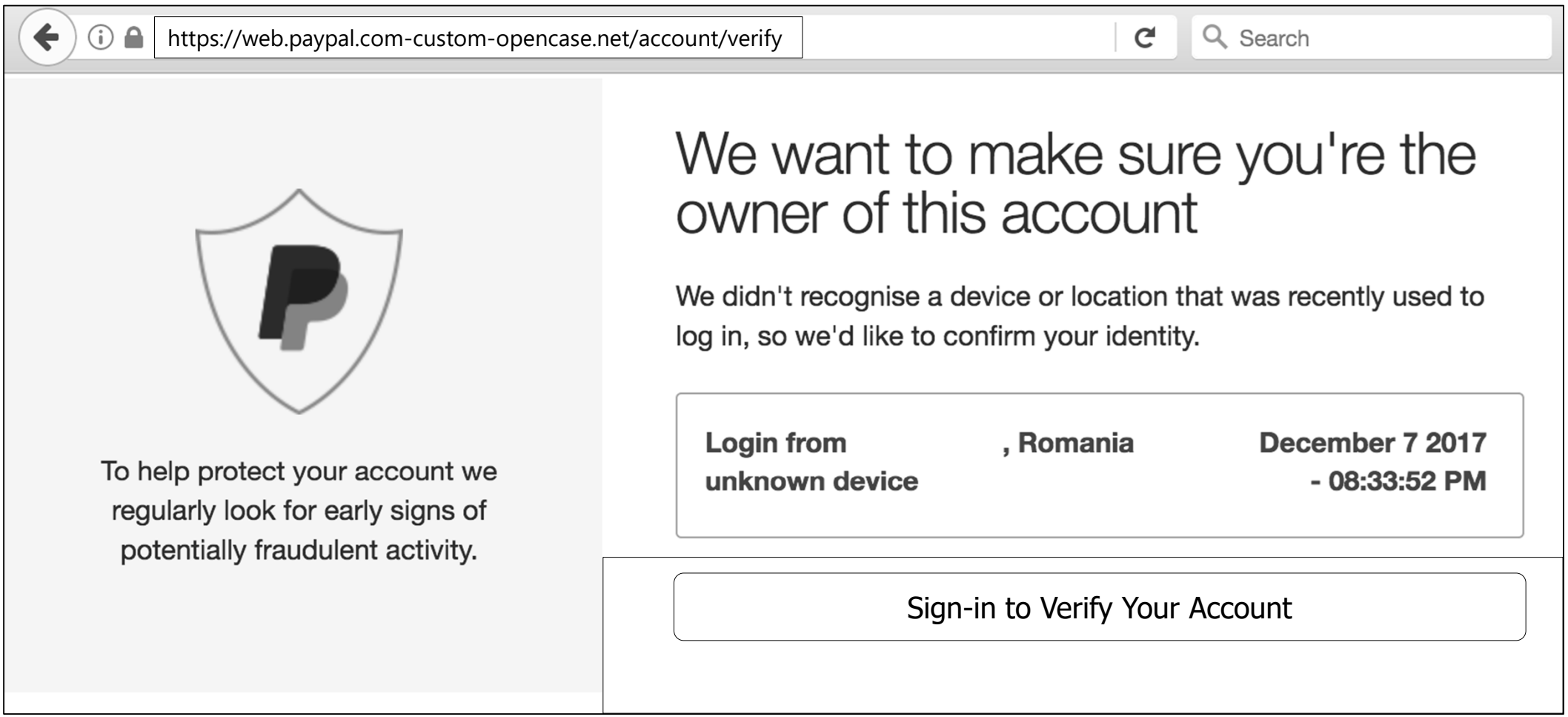
HTTPS extends HTTP by encrypting **data in transit** using **Transport Layer Security** (TLS, newer) or **Secure Sockets Layer** (SSL, older).



The initial DNS to resolve the domain is unencrypted, but afterwards:

- Request URLs including query string data is encrypted.
- Data in both the request **and** response are encrypted.
- Data is **encrypted in transit** and **decrypted at the client and server**.

What do you think? Is this page safe?



Examples of other application protocols

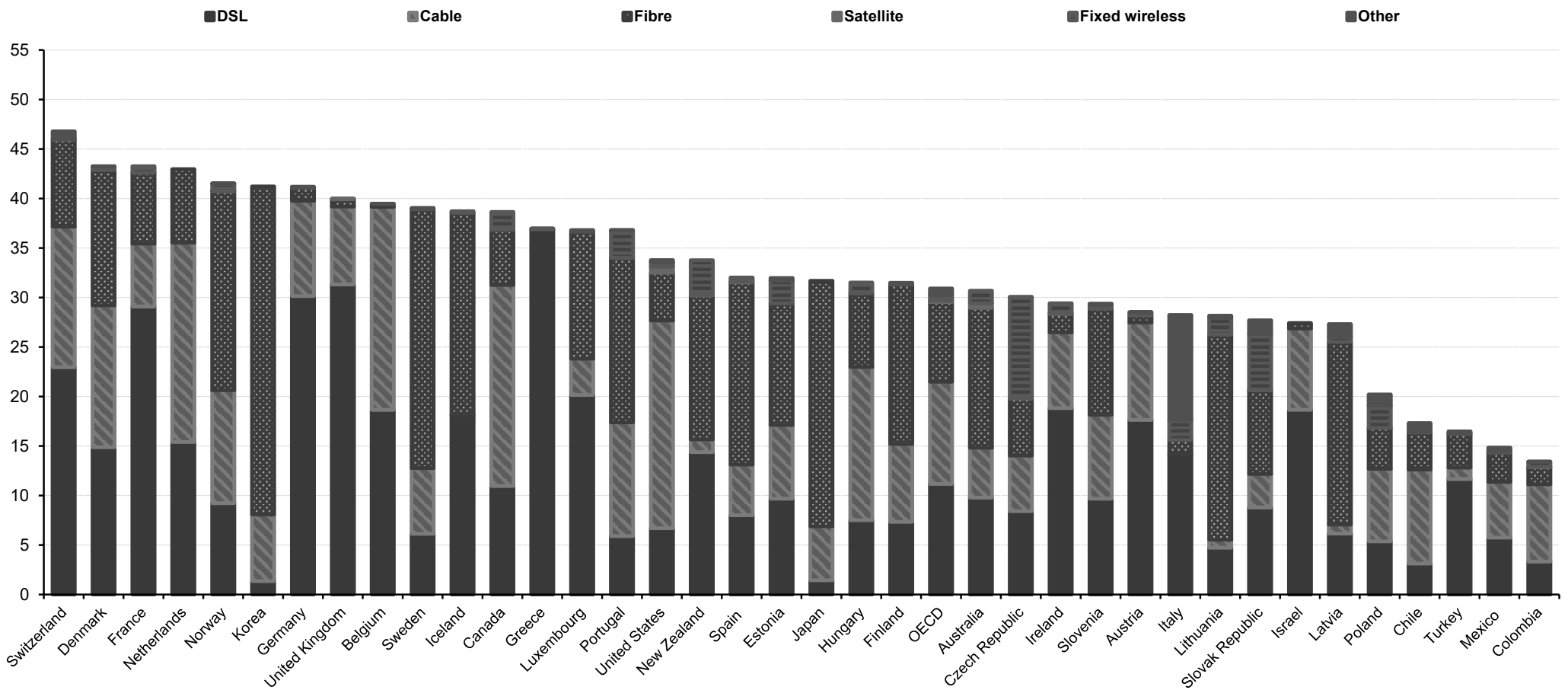
Protocol URL schemes	browser URI schemes	Application URI schemes
ftp		news
sftp	mailto	webcal
gopher	javascript	tel
telnet	data	facetime
imap	blob	skype
smtp	view-source	spotify
pop	about	steam
rtmp	chrome	git
rtsp	chrome-extension	bitcoin
Xmpp		magnet

Internet Connections

Connection links - many technologies

- PSTN - dialup, ISDN
- DSL (ADSL, ADSL2, ADSL2+)
- Cable (HFC)
- Fibre (FTTN, FTTC, FTTB, FTTP)
- Mobile (2G GPRS, 3G HSPA, 4G LTE, 5G NR)
- Wi-Fi (A, B, G, N, AC, etc.)
- Fixed wireless
- Microwave
- Satellite

Fixed broadband subscriptions per 100 inhabitants (Dec 2018)



http://www.oecd.org/sti/broadband/broadband-statistics/

Connection links - design issues

- **Link capacity**
  - Legal definition of "broadband" can vary by country
- **Variability in bandwidth capacity**
  - Symmetrical and asymmetrical capacities
- **Performance**
  - May not match theoretical or advertised capacity
- **The "digital divide"**
  - Affordability
  - Regional availability

Response times

Main factors influence perceived response times

- **Bandwidth** - the rate at which you send/receive data
- **Latency** - the time taken for data to get from Point A to Point B
  - May also be a result of processing delay at the server
- **"Stutter" or "Jank"** - perceived by the user
  - Usually caused by processing delay at the client end

New technologies increase bandwidth and decrease latency

- Increases the range of applications for which the web might be used

Response time expectations

- **0.1 seconds**
  - Feels like it was caused by the user, not the computer.
  - Essential to support the feeling of direct manipulation.
- **1 second**
  - Users can sense a delay, and thus know the computer is generating the outcome but they still feel in control.
- **10 seconds**
  - Users feel at the mercy of the computer and wish it was faster.
  - After 10 seconds, they start thinking about other things

Bandwidth

Technology	Capability	Typical Speeds	Notes
Dial-up	56 kbit/s	< 48 kbit/s	Rarely used
ADSL	8 Mbit/s	~ 6 Mbit/s	Varies by distance
ADSL2+	24 Mbit/s	~ 15 Mbit/s	Varies by distance
Cable	250 Mbit/s	< 100 Mbit/s	Shared bandwidth
Fibre	1 Gbit/s	~25-100 Mbit/s	Assuming FTTP
Satellite	1 Gbit/s	< 25 Mbit/s	Line-of-sight
2G GSM	237 kbit/s	~ 60 kbit/s	Rarely used
3G UMTS	14.4 Mbit/s	~ 5 Mbit/s	Highly variable
4G LTE	1 Gbit/s	~ 40 Mbit/s	Highly variable

Latency

Type of connection

- Wired connections typically have lower latency than wireless
- New technology typically has lower latency than older equivalents

Connection distance

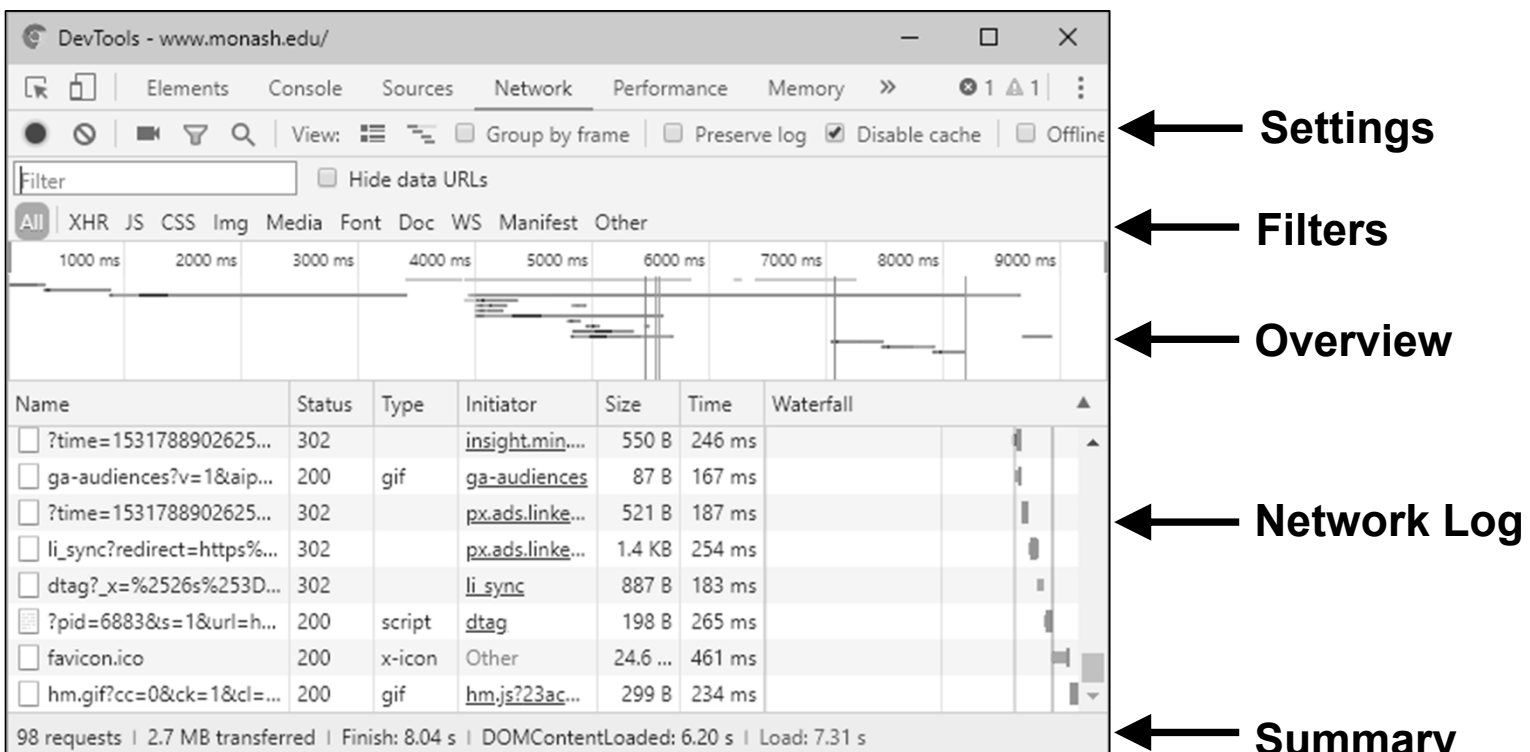
- Physically longer distances require more time to transmit.
- Longer distances require usually more “hops” between servers

Connection quality

- Transmission errors result re-routing over longer routes
- Transmission errors require data to be re-sent

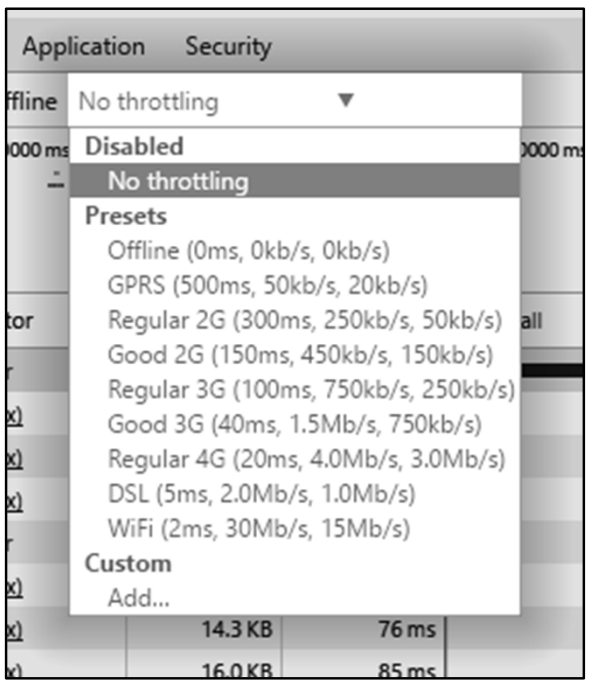
Test it yourself!

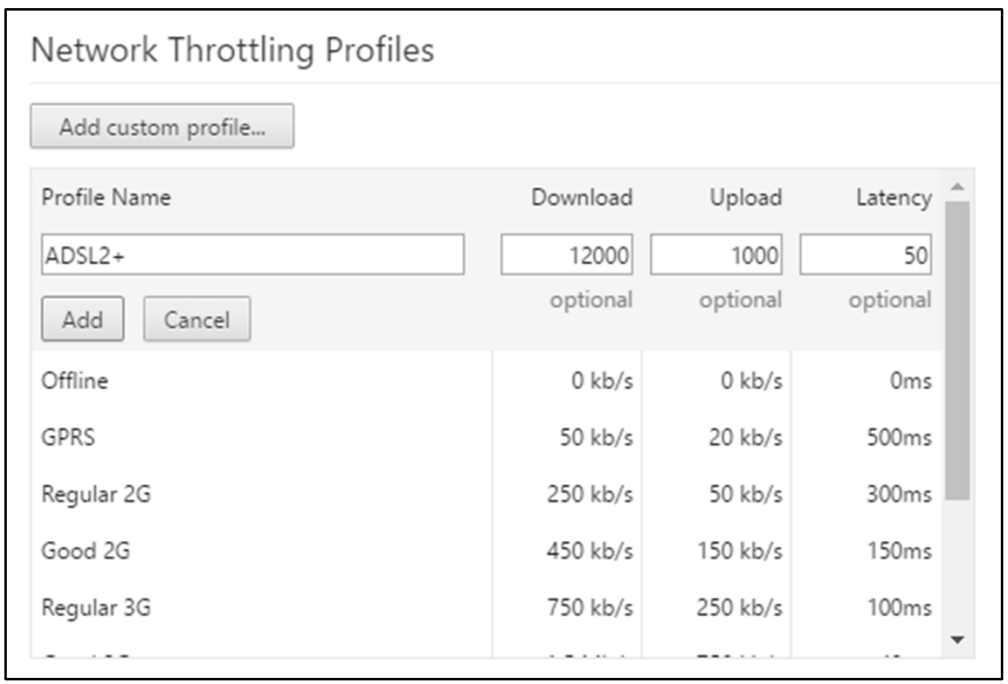
The developer tools in Google Chrome (and other browsers) includes a **network** panel that can visualise site loading performance:



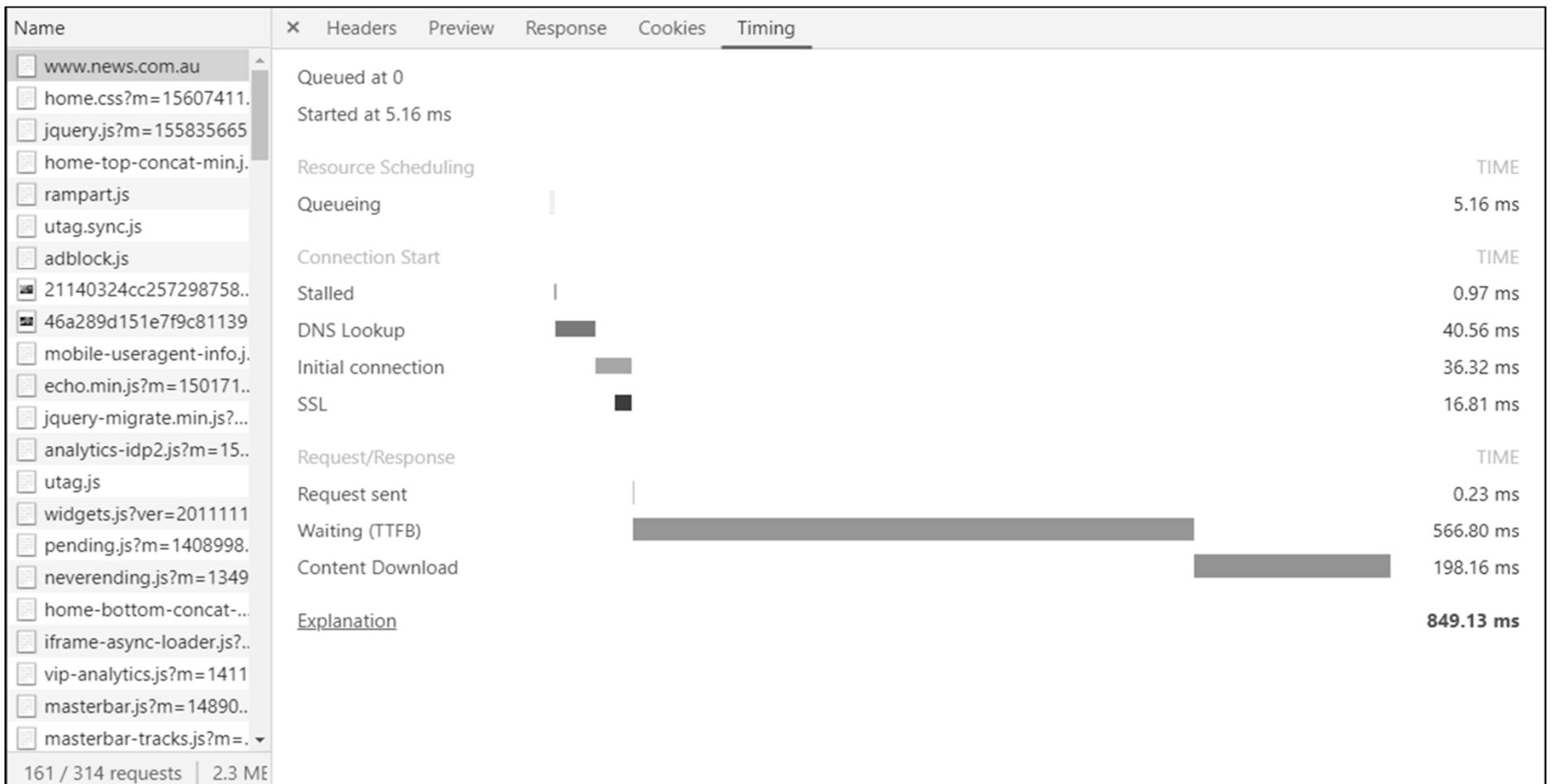
Simulating bandwidth and latency

The **network** panel in Chrome allows includes an option to artificially throttle loading of resources.





Not just file size!



Minimising bandwidth and latency

Optimise to transmit less data and reduce time spent waiting for data.

- Remove unnecessary files
- Prefer single files over many files
- Author media at appropriate dimensions
- Choose efficient file formats and encoding
- Compression - both file-based and HTTP compression
- Selectively pre-load or lazy-load content
- Use severs located closer to users

Content delivery networks

https://www.akamai.com/us/en/resources/visualizing-akamai/media-delivery-map.jsp

Reminders!

Participation milestone 1 is due this week. If you did not attend the lab session, contact your tutor as soon as possible!

Students studying in China:
Check Moodle for the online participation activity to be completed this week.

Homework:

- Begin researching your assigned article for assignment 1.
- Is there a technical issue that needs to be explained?
- Consider: What is the impact of the issue on the average person?