Curtin University

Application Layer II

Prof. Ling Li | Dr. Nadith Pathirage | Lecture 10

Semester 1, 2021

Curtin University

Application Layer Protocol - DHCP

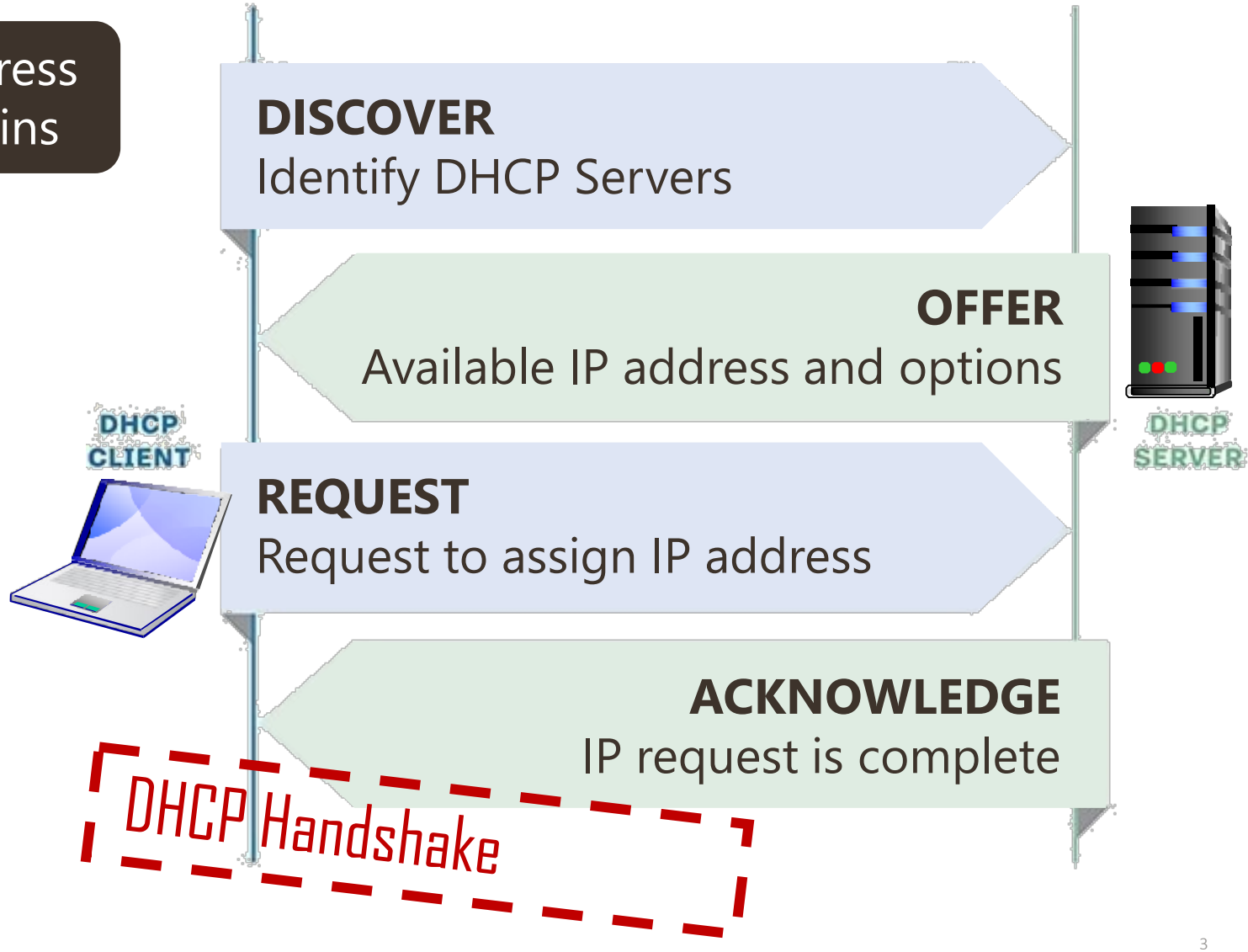
- Fundamentals
- 4-way Handshake
  - DHCP Discover
  - DHCP Offer
  - DHCP Request
  - DHCP Ack

DHCP: Dynamic Host Configuration Protocol

Curtin University

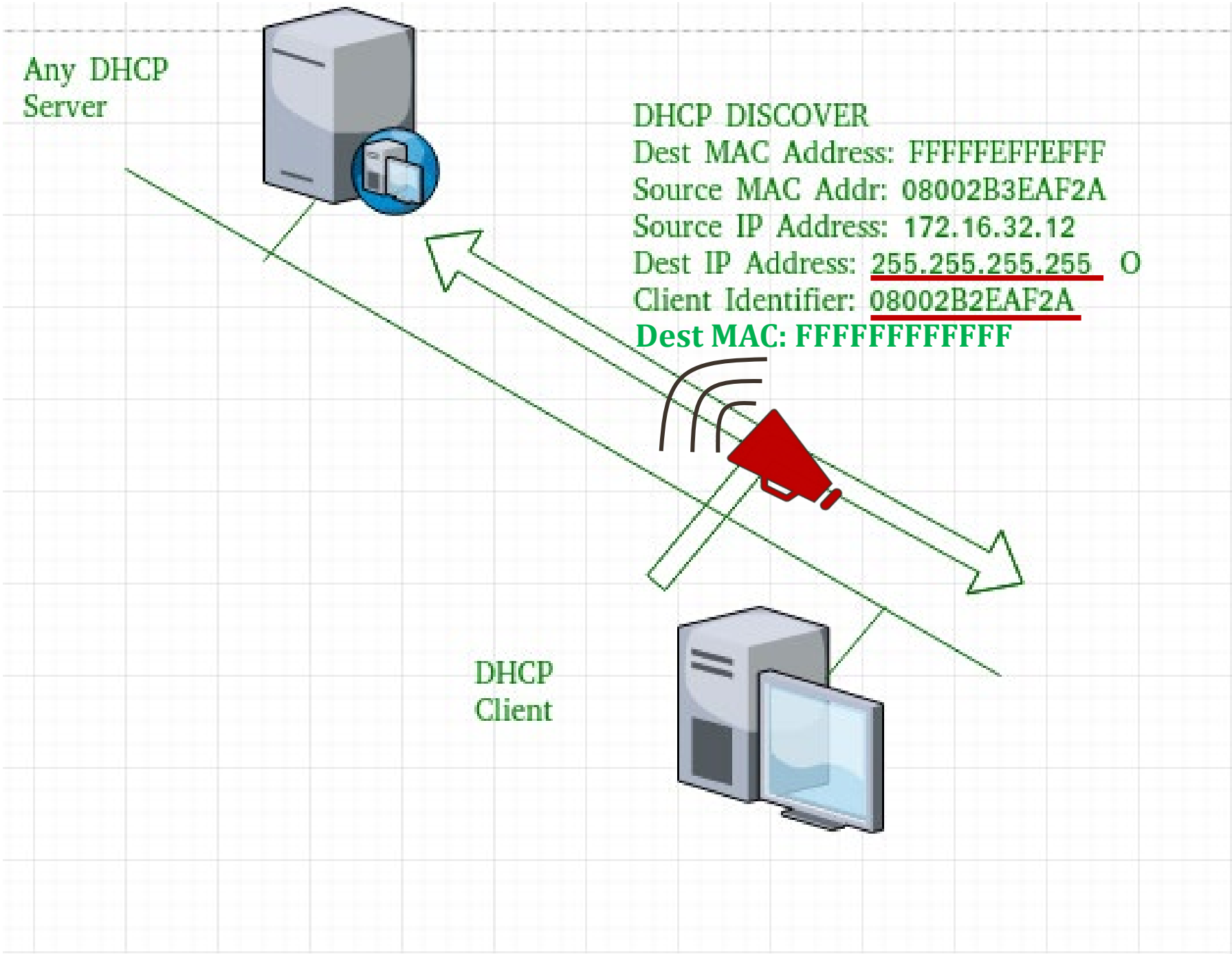
Goal: Dynamically obtain an IP address from network server when a host joins

- **Can renew** its lease on IP address in use
- **Allows reuse** of addresses (only hold address while connected /"on")
- Support for mobile users who want to join network (more shortly)
- Uses **UDP Broadcast**



1 DHCP Discover

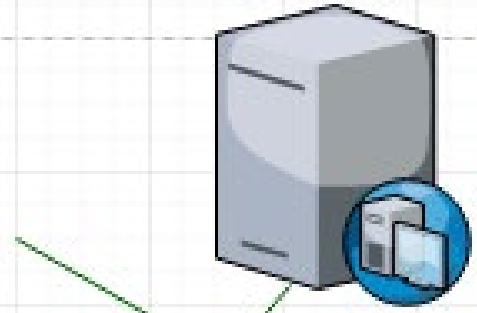
Generated by Client host **to discover any DHCP server/servers in a network**



2 DHCP Offer

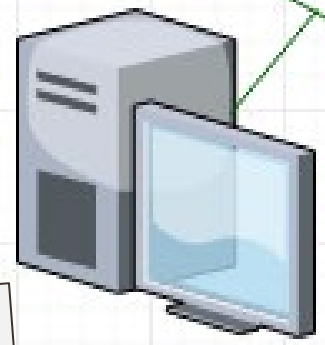
Server's Offer (UDP Broadcast)

- **Unleased IP Address**
- **Server Identifier:** To identify the server if there are many
- **Other TCP configuration** (Default GW, DNS server) information.



If many DHCP servers exists, accept the first DHCP OFFER

DHCPOFFER
Dest MAC Address: FFFFFEFFEFFF
Source MAC Addr: 00AA00123456
Source IP Address: 172.16.32.12
Dest IP Address: 255.255.255.255
Offered IP Address: 172.16.32.51
Server Identifier: 172.16.32.12
lease Length: 72 Hours
Client Identifier: 08002B2EAF2A



DHCP Client

Renew IP after 3 days

To identify the server if there are many

Other TCP Configuration

Curtin University

ipconfig /all

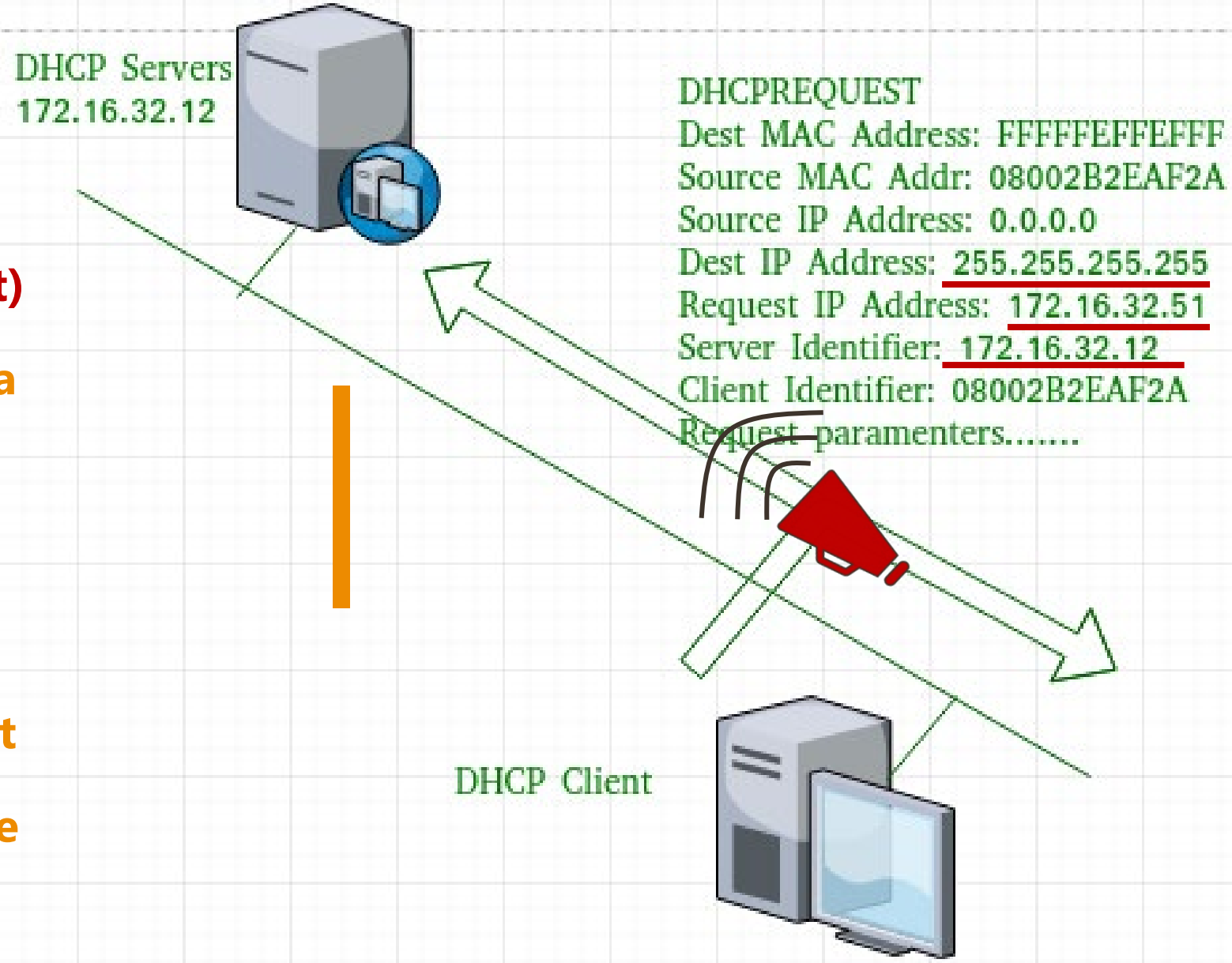
```
Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : stmary.local
   Description . . . . . . . . . . . : Intel(R) Dual Band Wireless-AC 8265
   Physical Address. . . . . . . . . : 88-B1-11-37-25-67
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::55d0:f8cb:699e:fa1b%18(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.172.112.109(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.248.0
   Lease Obtained. . . . . . . . . . : Monday, March 26, 2018 8:35:05 PM
   Lease Expires . . . . . . . . . . : Wednesday, March 28, 2018 7:19:47 PM
   Default Gateway . . . . . . . . . : 10.172.112.1
   DHCP Server . . . . . . . . . . . : 172.16.13.12
   DHCPv6 IAID . . . . . . . . . . . : 143175953
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-21-2D-EC-36-54-E1-AD-5D-00-2F
   DNS Servers . . . . . . . . . . . : 172.16.13.12
                                       172.16.13.10
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

3 DHCP Request

Client request (UDP Broadcast)

- **Network configuration data including an IP address**
- **If more than one offers were received:**
  - **Select one and indicate the server in the request**
  - **Inform all servers on the selection**

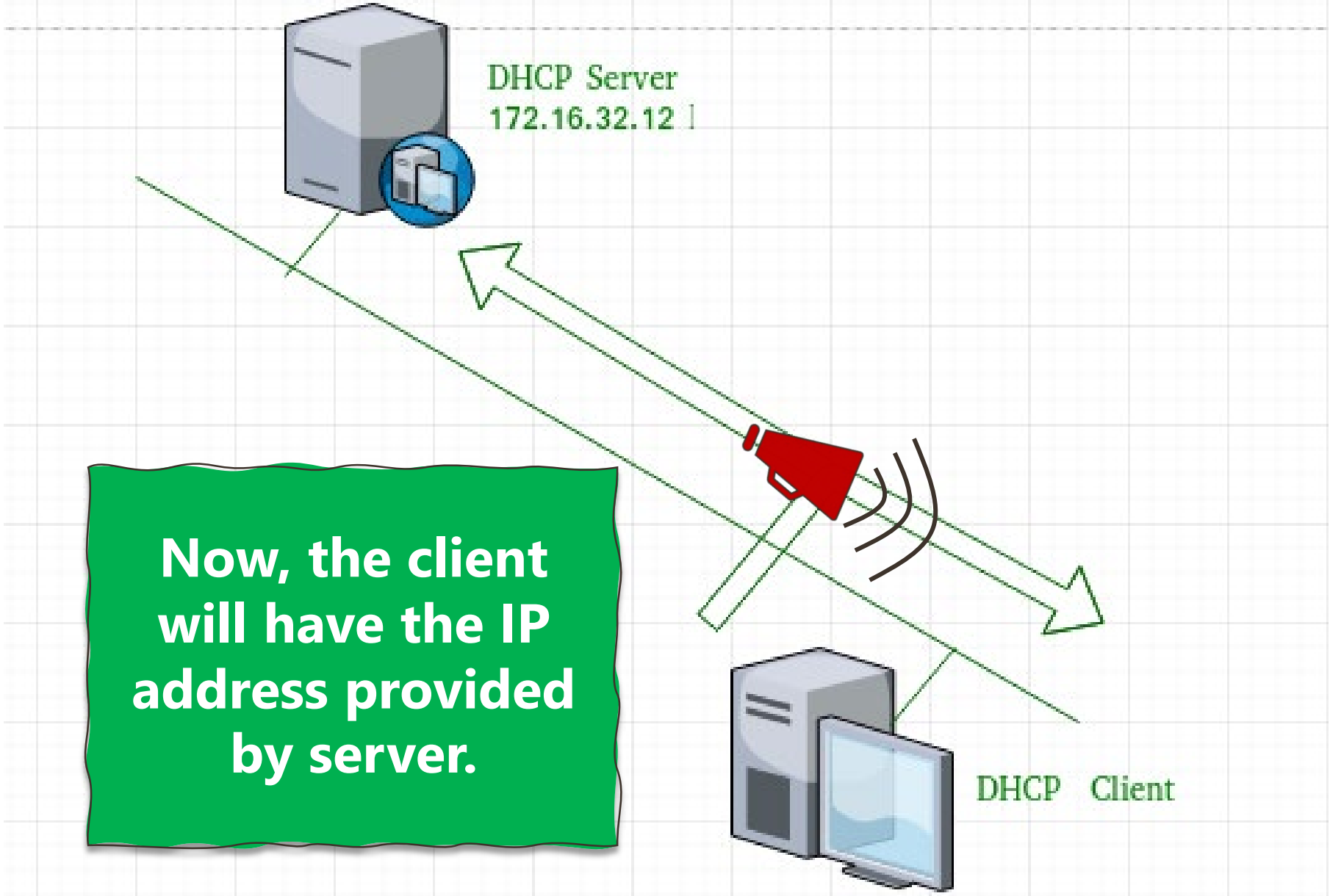
"showing the acceptance of IP address"



4 DHCP Ack

Server will make:

- **Entry with specified client ID**
- **Bind the IP address** offered **with lease time**



Curtin University

Domain Name System

- Domain Name System (DNS)
  - hosts File
- Main Elements
  - Domain Name Space
  - Name Server
  - Resolvers
- DNS Database
- DNS Registrar
  - Web Hosting

DNS

Curtin University

- Application program refers to **host by ASCII** string names:

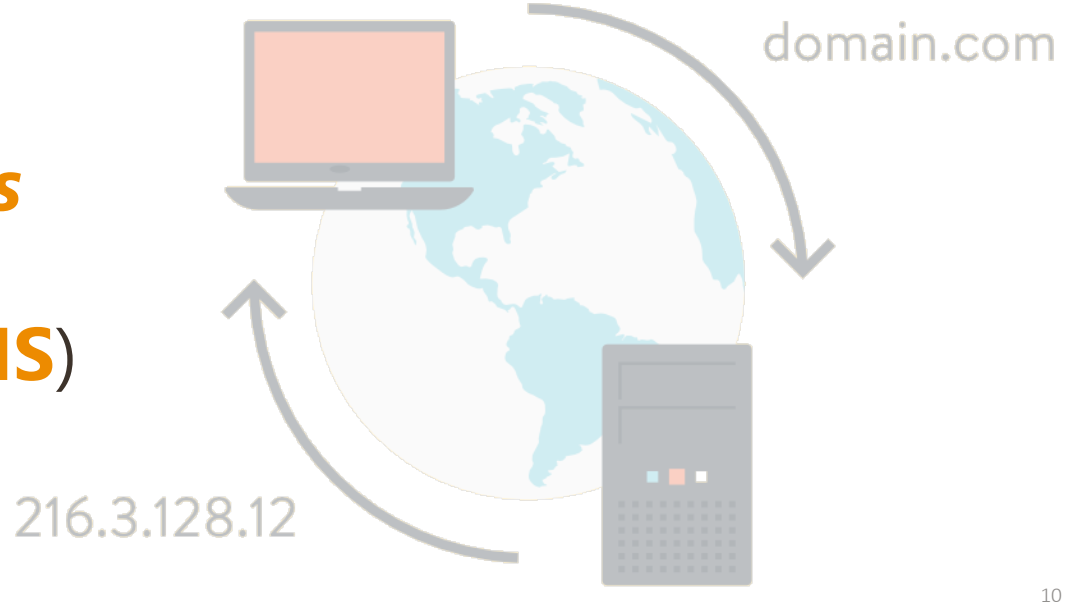
  name: ark.cs.curtin.edu.au

  IP address: 134.7.1.10

- Network does not understand ASCII names

- Given a **machine name** - *need some mechanism to* ***convert to an IP address***

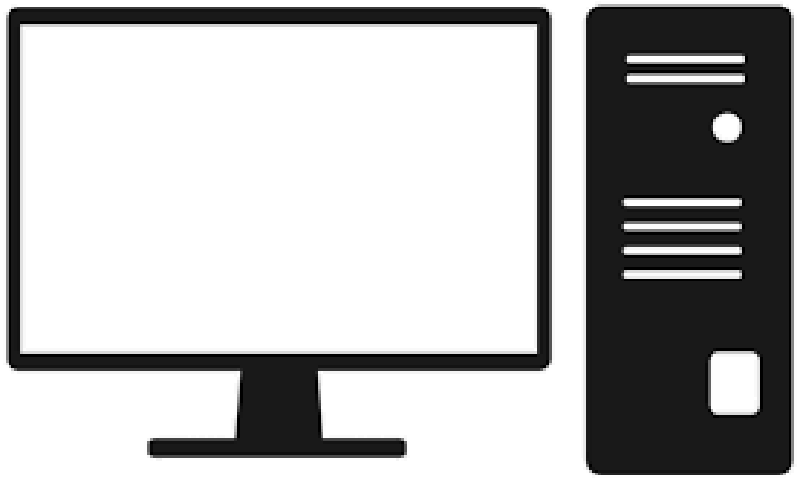
- **Today:** uses Domain Name System (**DNS**)
  - ✓ Defined in RFC 1034 and RFC 1035
  - ✓ A directory lookup service



Hosts File

Curtin University

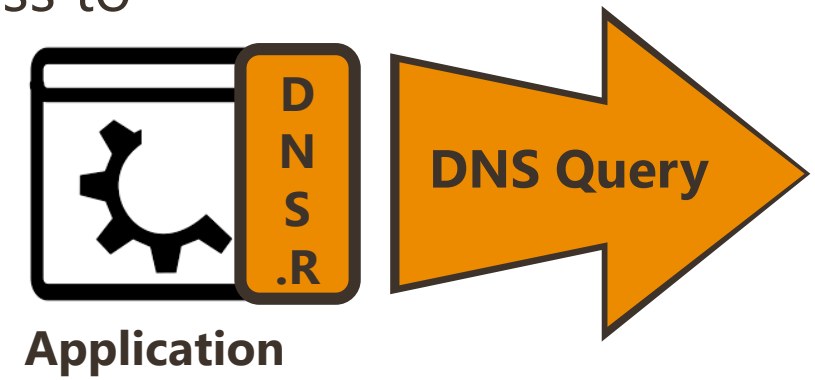
- Path
  "C:\Windows\System32\drivers\etc**hosts**"
- **Plain text file**
- **Maps hostnames** to **IP addresses**
- Originally a file named HOSTS.TXT was manually maintained for ARPANET



DNS – Cont.

Curtin University

- **DNS plays a support role to other applications**
  - Application program calls a library procedure – the **resolver** to resolve a remote host name.
  - **Resolver** then queries the local **DNS server**
  - DNS server answers the query with an IP address
  - Application program can then use the IP address to communicate with the remote host
  - All DNS queries **use UDP**



DNS Main Elements

Curtin University

1. **Domain Name Space**
2. **Name Servers**
3. **Resolvers**

1. Domain Name Space

Curtin University

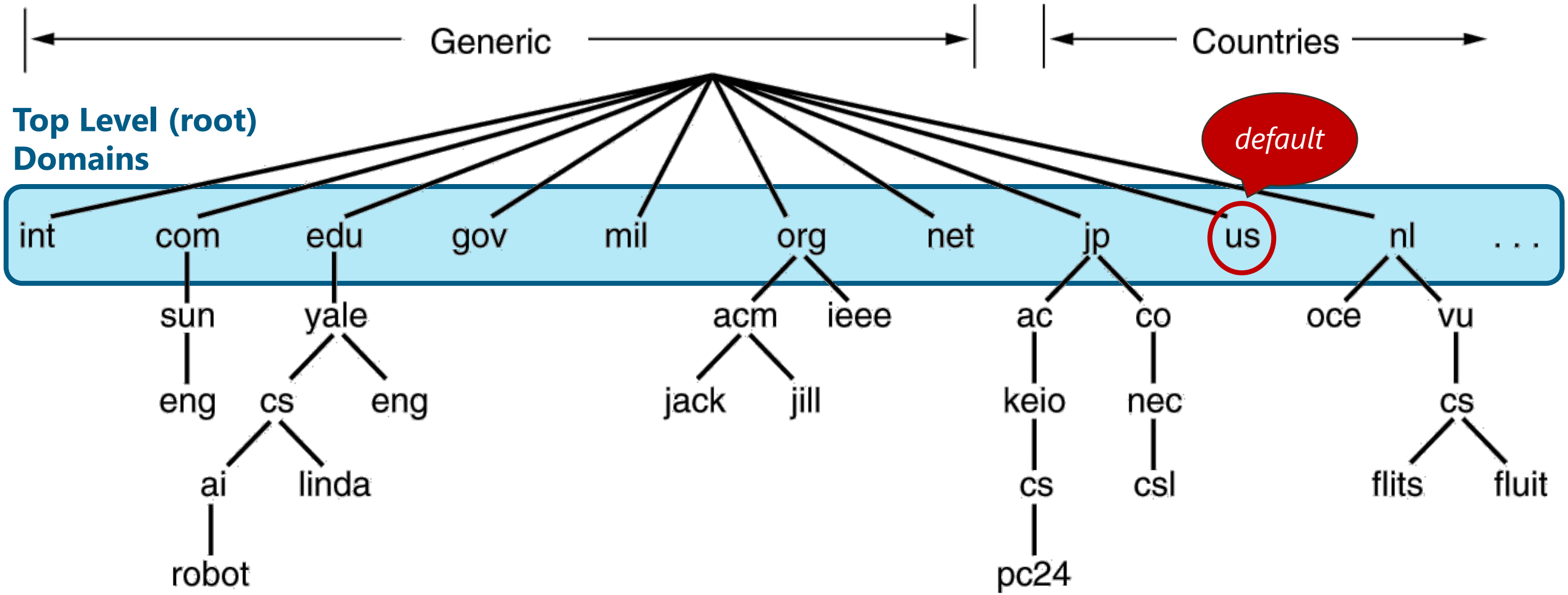
- **DNS** uses a **hierarchical, domain-based naming scheme** to identify resources on the Internet.

- At the top are a small number of domains that **encompass the entire internet**

  ✓ Controlled by **ICANN**
  (International corporation for assigned names and number)

Internet Domain Name Space

Curtin University

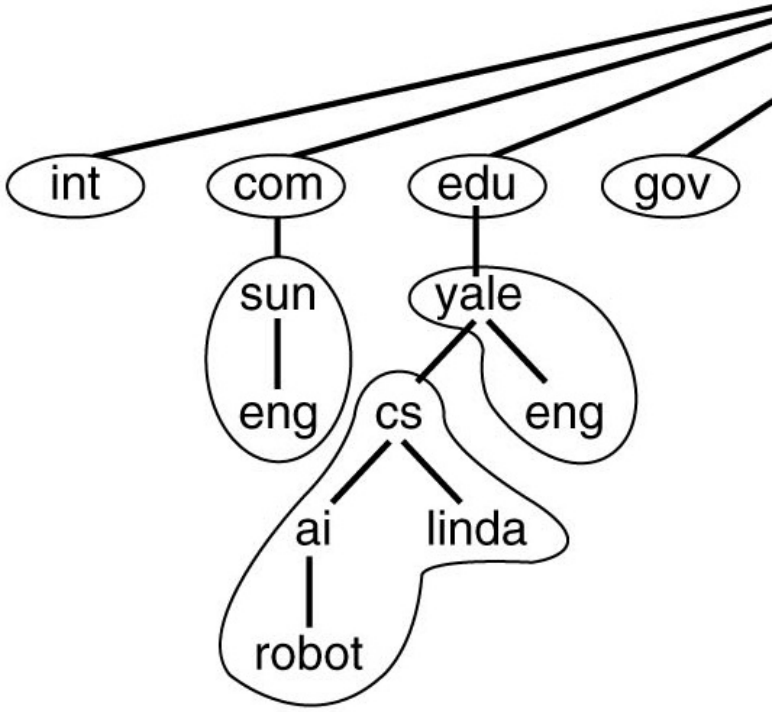


robot.ai.cs.yale.edu.us

1. Domain Name Space – cont.

Curtin University

- Domain names are **case insensitive**,
  - *Edu, edu, EDU*
- Component names can be up to 63 characters long and the full path cannot exceed 255 characters
- Naming follows **organizational boundary**, **not physical network!**
  - One domain may consist of multiple subnets & vice versa



2. Name Servers

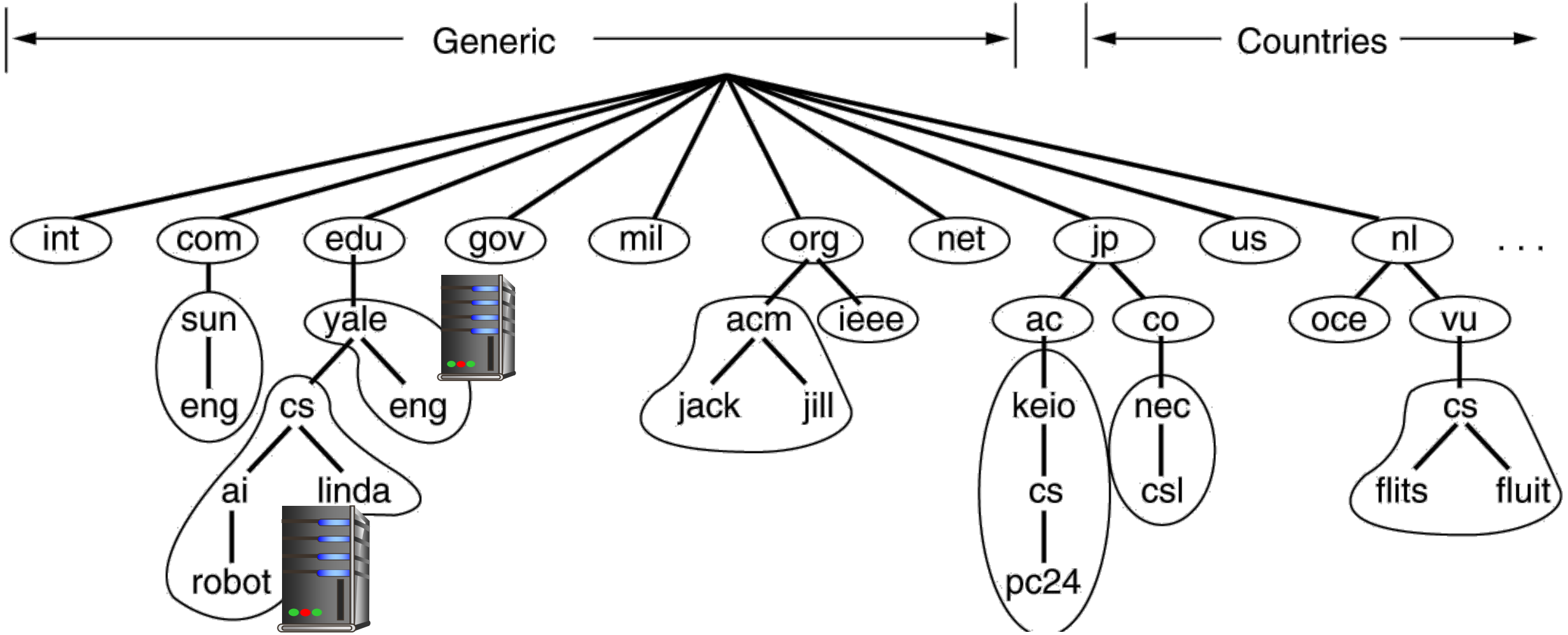
Curtin University

- **Distributed Approach:** dividing into **non-overlapping "zones"**
- **Each Zone:**
  - **Contains some part of the DNS tree**
  - Managed by **one or a set of DNS servers** – normally one primary server or/and one or more secondary servers
  - **Managing server** does not have to belong to the zone it manages
  - **Zone boundaries** are the decision of the **zone administrator**

For Example:
Curtin zone, managed by Curtin DNS server,
while Computing zone, managed by Computing DNS server

DNS Zones

Curtin University



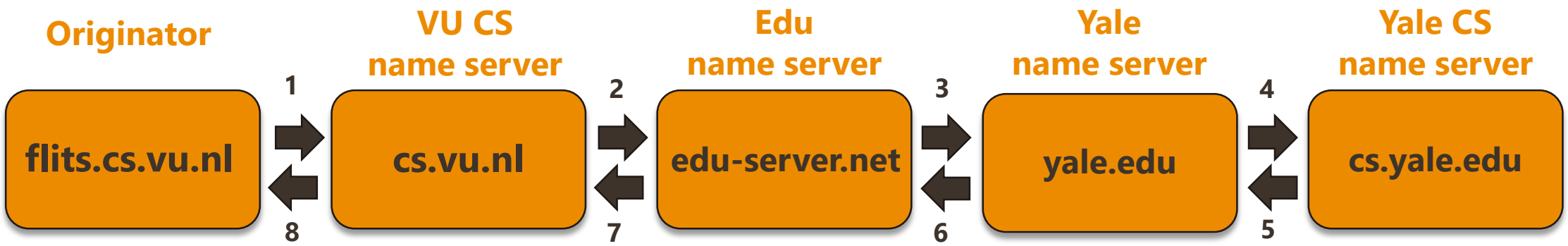
3. Resolvers

Curtin University

- DNS server receives a **type A query** from a resolver (client) for **"www.mit.edu"**
  - **If** domain name being queried is managed by the server, returns the matched RR **(authoritative record)**
  - **If** a matched RR exists on the cache, returns the cached RR **(non-authoritative answer)**
  - **Else,**
    - (recursive query method) **sends a query to** the DNS server for the **top-level domain "edu"**, waits for answers, and forward answers back to client.

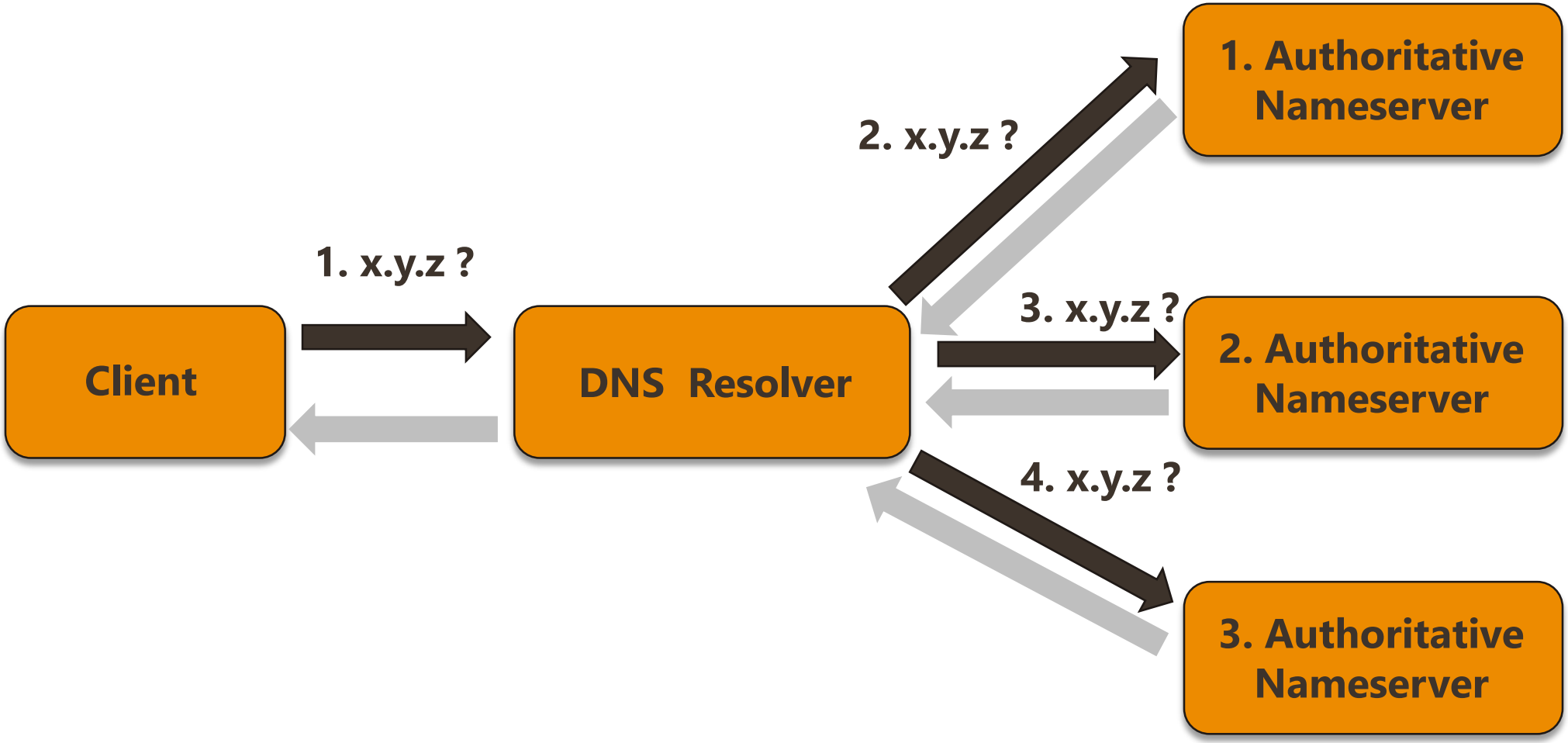
Resolver – Implementation 01

Curtin University



Resolver – Implementation 02

Curtin University



DNS Database

Curtin University

- DNS is based on a hierarchical database containing **Resource Records (RRs)**
- Every domain **can have a set of RRs**
- A RR contains – the host name, the IP address and other information about the host
- **Structure** of RR format:

Domain Name	Time To Live	Class	Type	Value
uranus.cs.curtin.edu.au	86400	IN	**A**	**137.7.2.130**

RR Types

Curtin University

Domain Name	Time To Live	Class	Type	Value
uranus.cs.curtin.edu.au	86400	IN	TXT	"Lab219 Computing Curtin"
uranus.cs.curtin.edu.au	86400	IN	**A**	**137.7.2.130**
uranus.cs.curtin.edu.au	86400	IN	**A**	**137.7.1.112**
uranus.cs.curtin.edu.au	86400	IN	**MX**	**1 bike.cs.curtin.edu.au**
uranus.cs.curtin.edu.au	86400	IN	**MX**	**2 dns.cs.curtin.edu.au**
uranus.cs.curtin.edu.au	86400	IN	HINFO	Redhat Linux 7.1
www.cs.curtin.edu.au	86400	IN	CNAME	kickit.cs.curtin.edu.au

A:	Domain to IP
MX:	Mail Server
HINFO:	Host Info
CNAME:	Canonical Name

DNS SOA (Stat of Authority) RR

Curtin University

- Indicates which Domain Name Server (DNS) is the best source of information for the specified domain.
  - **Server Name:** Primary name server for the domain.
  - **Mailbox:** Email for the domain.
  - **Refresh time:** The number of seconds before the zone refreshes.
  - **Retry time:** The number of seconds before a failed refresh is retried.
  - **Expiration time:** The time, in seconds, before the data is considered unreliable.
  - **Minimum TTL:** The default that applies to all of the resource records in the zone.

Every domain must have an SOA record

Domain Name	Type	Value
oasis.com	SOA	Server Name: dns.oasis.com Mailbox: mail.oasis.com MinTTL: 100 Retry Time: 100 Refresh Time: 100 Expiry Time: 100

Inserting records into DNS

Curtin University

- **Example: new startup "Network Utopia"**
- Register name networkuptopia.com at **DNS registrar** (e.g., Network Solutions)
  - provide names, IP addresses of authoritative name server (primary and secondary)
  - **DNS registrar inserts two RRs** into **.com TLD server**:
    (networkutopia.com, dns1.networkutopia.com, NS)
    (dns1.networkutopia.com, 212.212.212.1, A)
- **create** authoritative server **type A record** for www.networkuptopia.com; type **MX record** for networkutopia.com, etc.

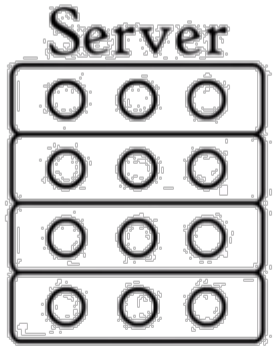
Website Hosting on Internet

Curtin University

1

Web Hosts ($)

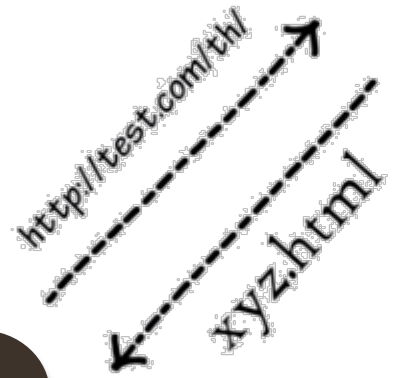
GoDaddy
HostGator
Bluehost



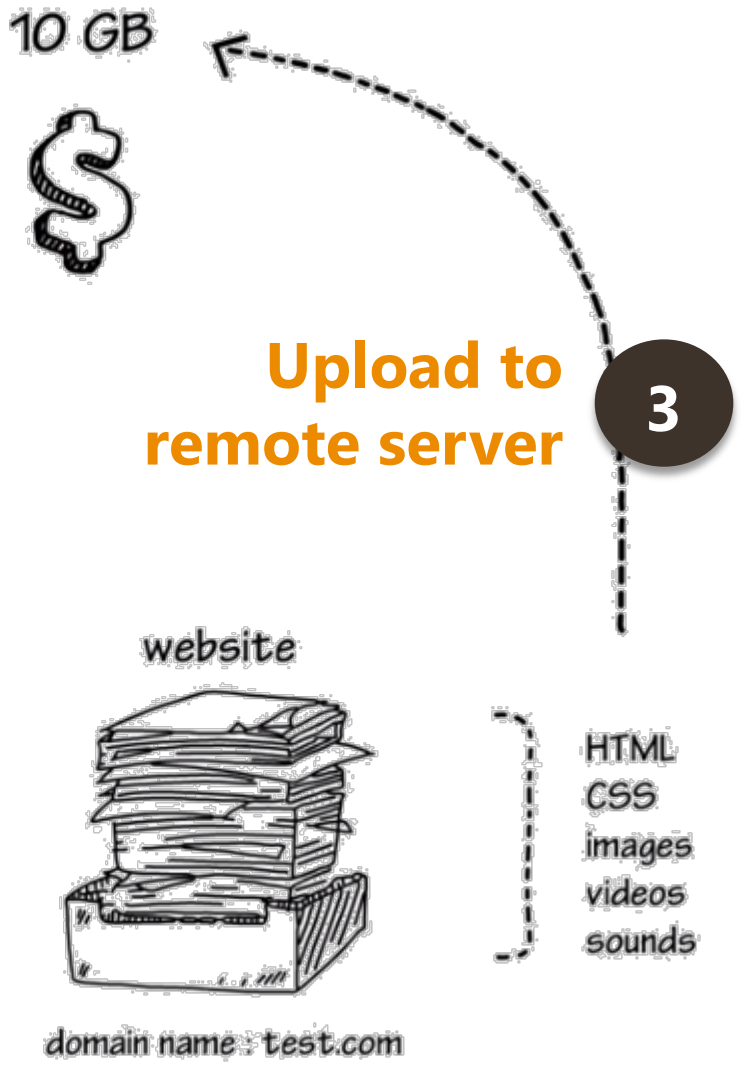
2

Domain Name Registrar ($)

1. Purchase an available domain name (www.test.com)
2. Set DNS entries for NS (Name Server)



4



Curtin University

Web
Search Engines

- Fundamentals
- Ranking Algorithm
- Search Results

Search Engines (SE)

Curtin University

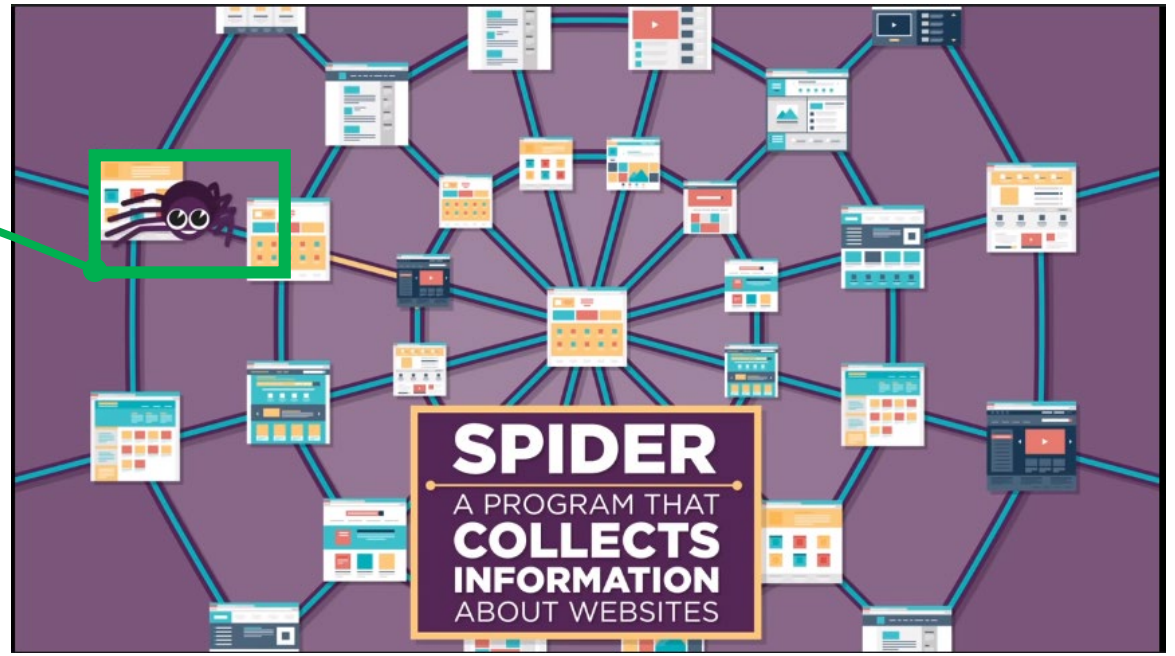
Request ->
Results ?
How ?

- **No real-time search in WWW**
  (There is > billion websites in WWW)
- SE constantly **scans** the **web in advance**

WWW is a web of pages connected (hyperlinks)

A **web spider (program)** travels through these links collecting information

Builds the **Search Index Database**



Search Engines: Ranking Algorithm

Curtin University

- **Search Query -> Results ? How ?**
  - Checks whether search terms shows up in page title
  - Sequence of words
  - Meta data of the website
- **Page Rank** (Invented by Google - Larry Page)
  - Considers how many other webpages linked to a given page
- SE regularly updates the Algorithm to avoid spamming (fake and untrustworthy site)
- **SEO: "Search Engine Optimization"** techniques would help to rank a page higher

If many webpages are interest on a page, then it's probably the one you are looking for.

Search Engine: Results

Curtin University

To provide better/faster results

- Constant updates to the algorithms
  - Use user's implicit information

(GEO location, Past search queries etc.)

Modern Search Engines are Intelligent

- Understand underlying meaning of the words
- Understand Images (Machine Learning)
- Understand Video Sequences (AI)

Curtin University

Web Services

- REST API
- URI / URL / URN
- REST API Highlights

Web Services

Curtin University

A web service is a software system designed to support interoperable machine-to-machine interaction over a network

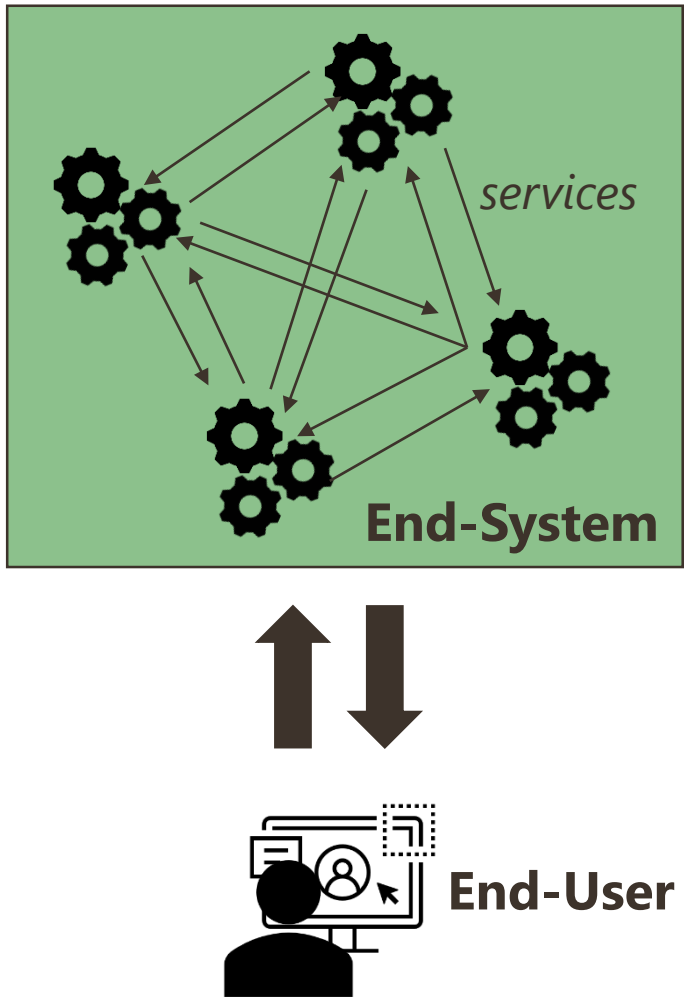
- **A Web Service** is **defined by rules**:
  - How software component will talk?
  - What kind of messages will be passed?
  - How requests and Reponses are handled?

- **Harness** the power of **HTTP**

Web Services: Why we need them?

Curtin University

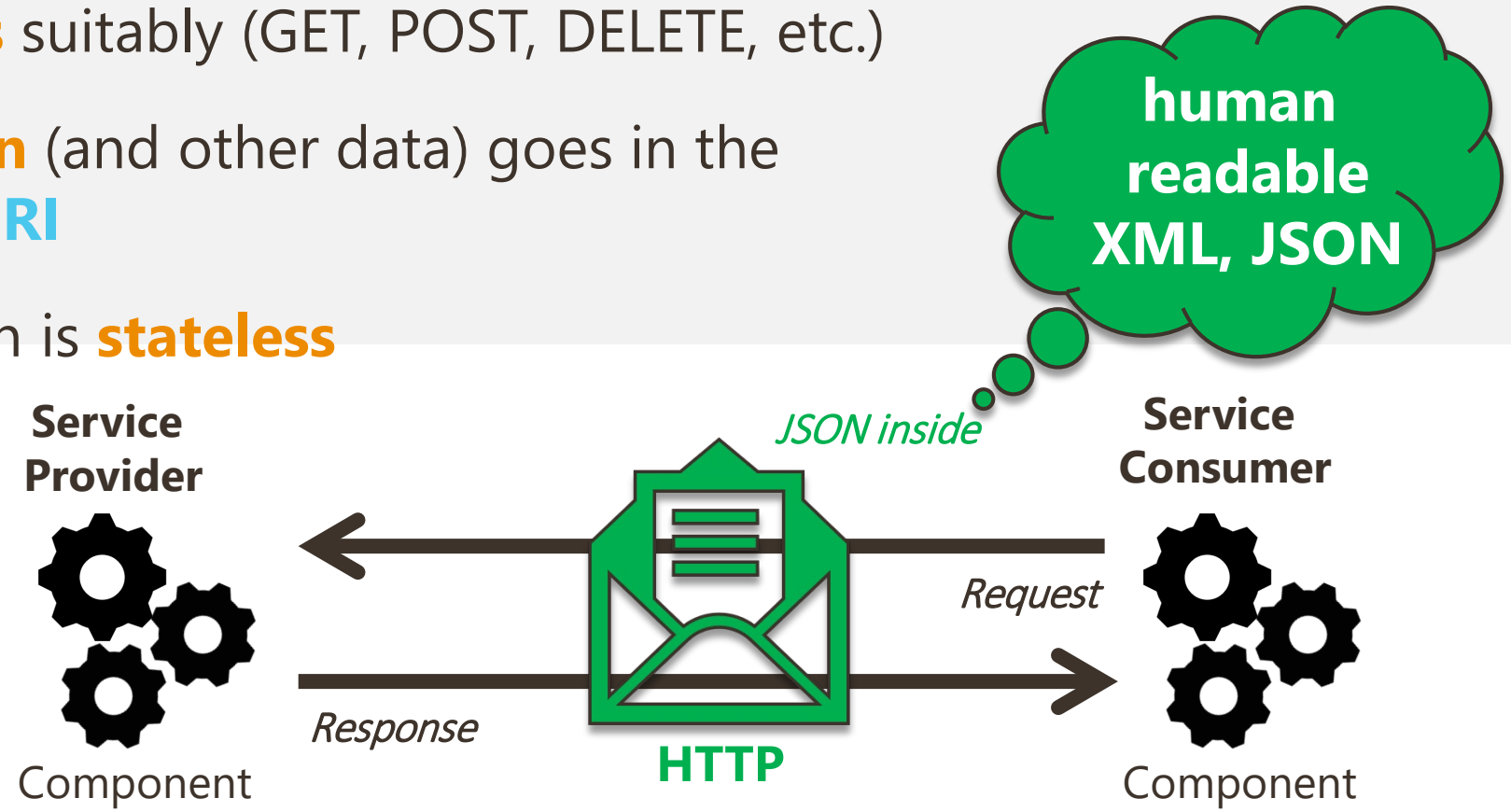
- **Distributed Programming**
  - allow various applications to talk to each other and share data and services among themselves
- The web is now a very large interlinked set of systems
- Allow component on the web to talk to each other
- Provide better user experience
- Associated with **Service-oriented Architectures (SOA)**
- SOAP (Simple Object Access Protocol) are commonly used (which relies on XML)



REST Web Service / REST API

Curtin University

- Latest is **RE**presentation **S**tate **T**ransfer **(REST)** Protocol
  - an architectural style that makes use of existing and widely adopted technologies, specifically HTTP
- Uses **HTTP methods** suitably (GET, POST, DELETE, etc.)
- **Scoping information** (and other data) goes in the **parameters of** the **URI**
- **REST** Communication is **stateless**



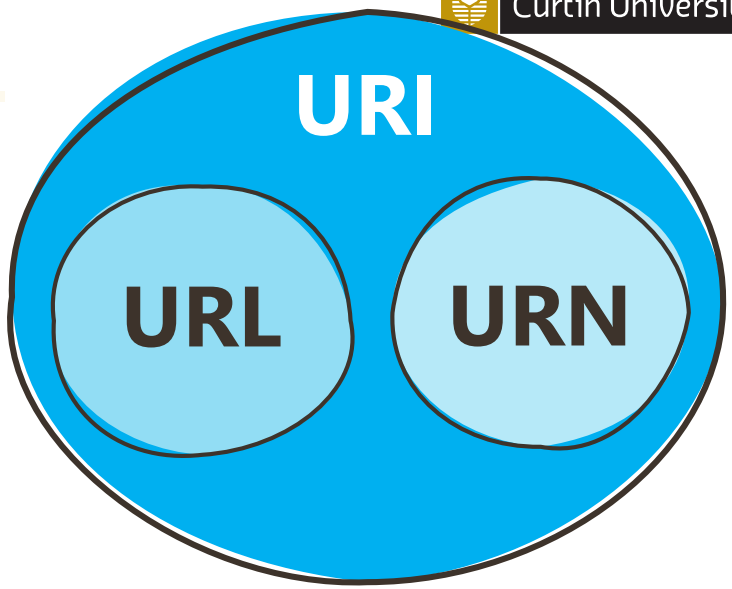
URI / URL / URN

Curtin University

Name:
Flink SideSmack
Address:
12, Beveridge St,
Bentley

Uniform Resource Identifier

- String of characters used to identify resource on the internet,
- **by location** or **by name**, or **both**



Uniform Resource Locater

Can contain **Query Strings**
http://www.myweb.com/signup?
fn=Flick&ls=Sidemack

Can contain **Fragments**
http://en.wikipedia.org/wiki/
web_developer**#external _links**

Uniform Resource Name

Subset of URIs that include a **name within a given space**, but no location

Name:
urn:isbn:0451450523

Curtin University

Peer To Peer

P2P

- Fundamentals
- Napster
- Gnutella
- BitTorrent – in depth
- P2P Web

What is peer-to-peer (P2P)?

Curtin University

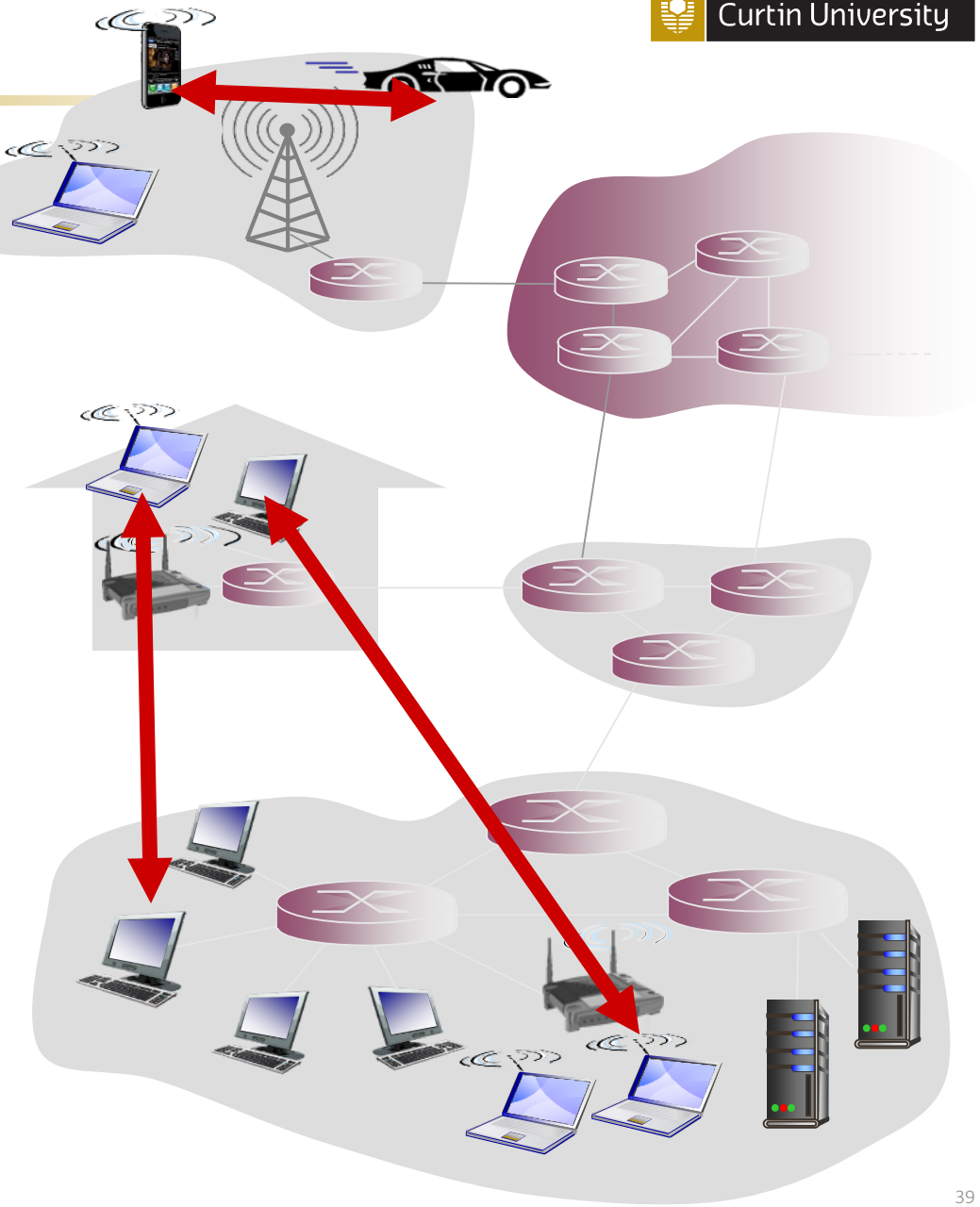
"Peer-to-peer is a way of structuring distributed applications such that the **individual nodes have symmetric roles**. Rather than being divided into clients and servers each with quite distinct roles, in P2P applications **a node may act as both a client and a server**."

-- Charter of Peer-to-peer Research Group, IETF/IRTF, June 24, 2004

Pure P2P Architecture

Curtin University

- **No always-on** server
- Arbitrary end systems **directly communicate**
- Peers are intermittently connected and **change IP addresses**

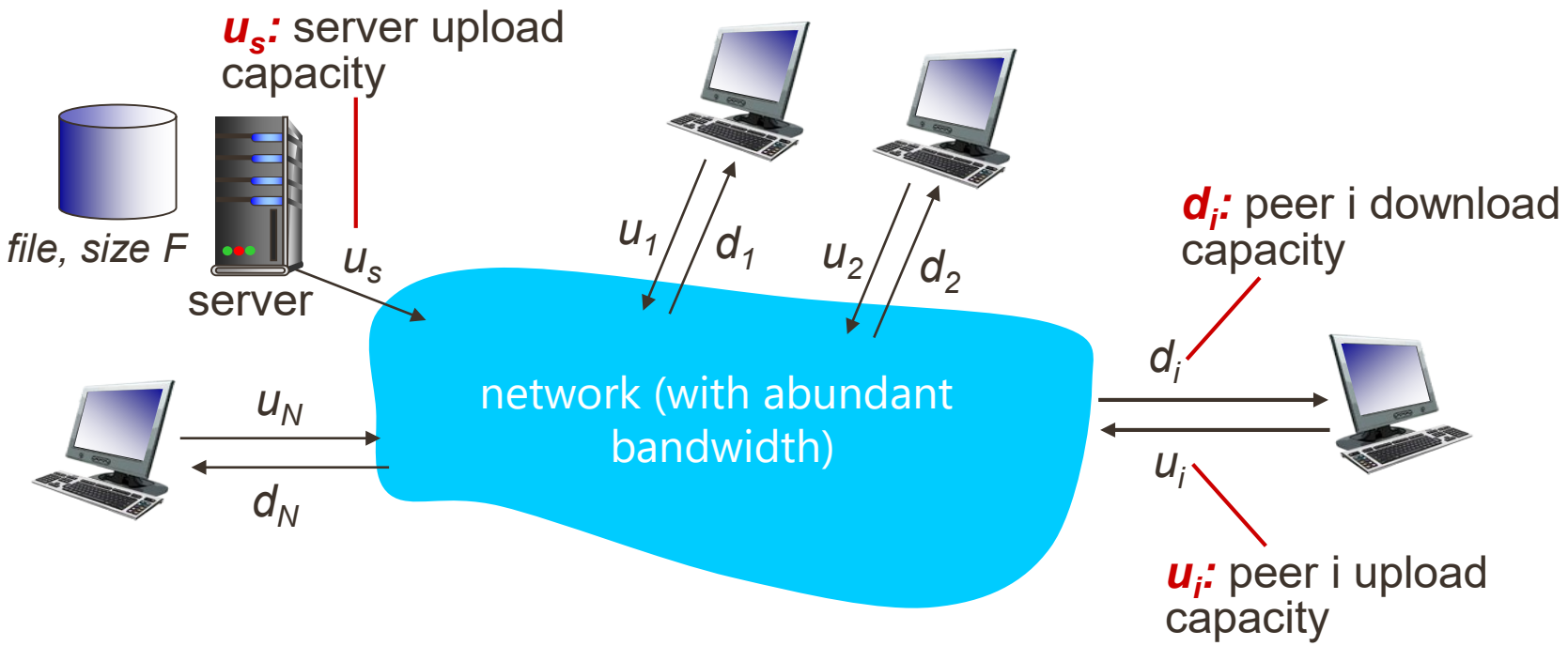


Client-server vs. P2P

Curtin University

Question: how much time to distribute file (size F) from one server to N peers?

- peer upload/download capacity is limited resource



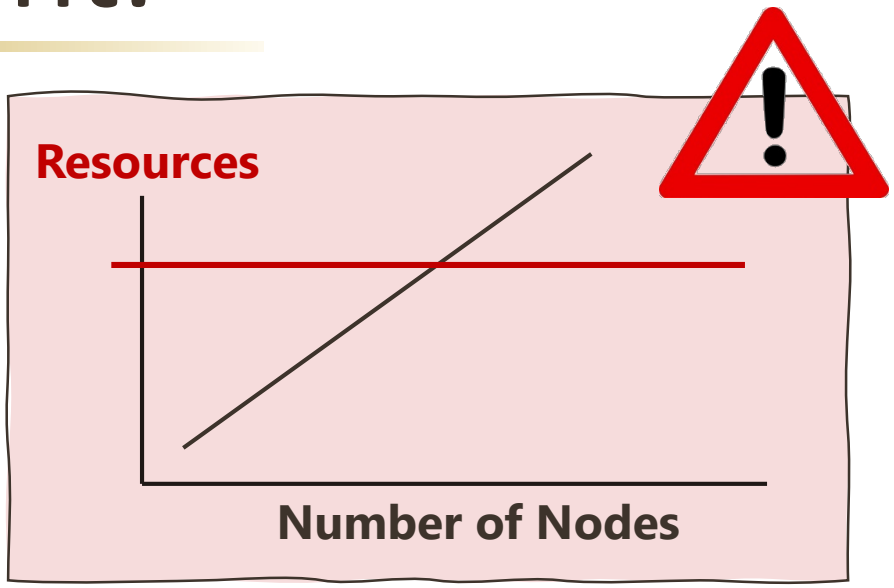
Client-server vs. P2P – cont.

Curtin University

- **Client-Server architecture**

  Problems:

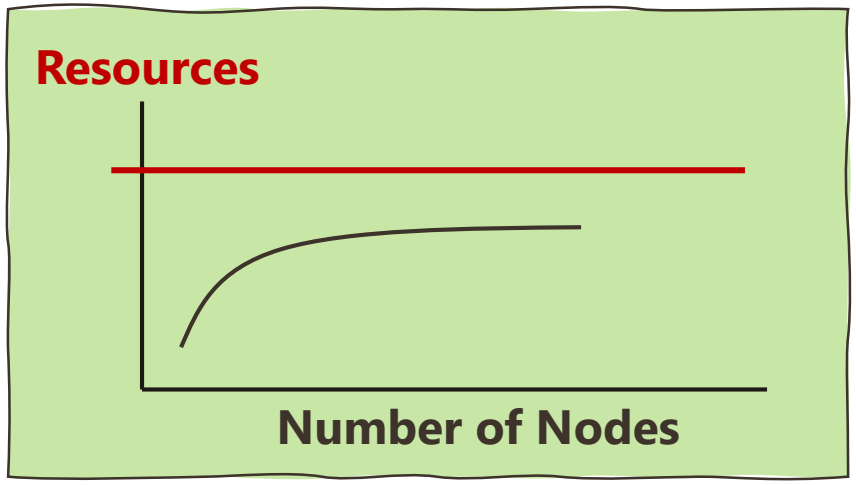
  - Limited resources
  - All loads are centered on the server
  - Server-based architecture has low scalability
  - The setup and maintenance cost is high



- **Peer-to-Peer (P2P) architecture**

  Advantages:

  - Distributing loads to all users
  - Users consume and provide resources
  - P2P architecture has high scalability
  - The setup and maintenance cost is low



P2P Protocols

Curtin University

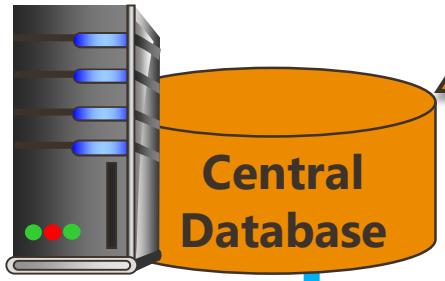
Hybrid
(not pure P2P)

Napster

hybrid system

NOT pure P2P network

All Clients send **ID's of the data** they hold to the server and when client asks for data, server responds with specific addresses



1. Query

2. Response
List of Clients

3. Download Request

4. File

Peer downloads directly from other peer(s)

Napster

Curtin University

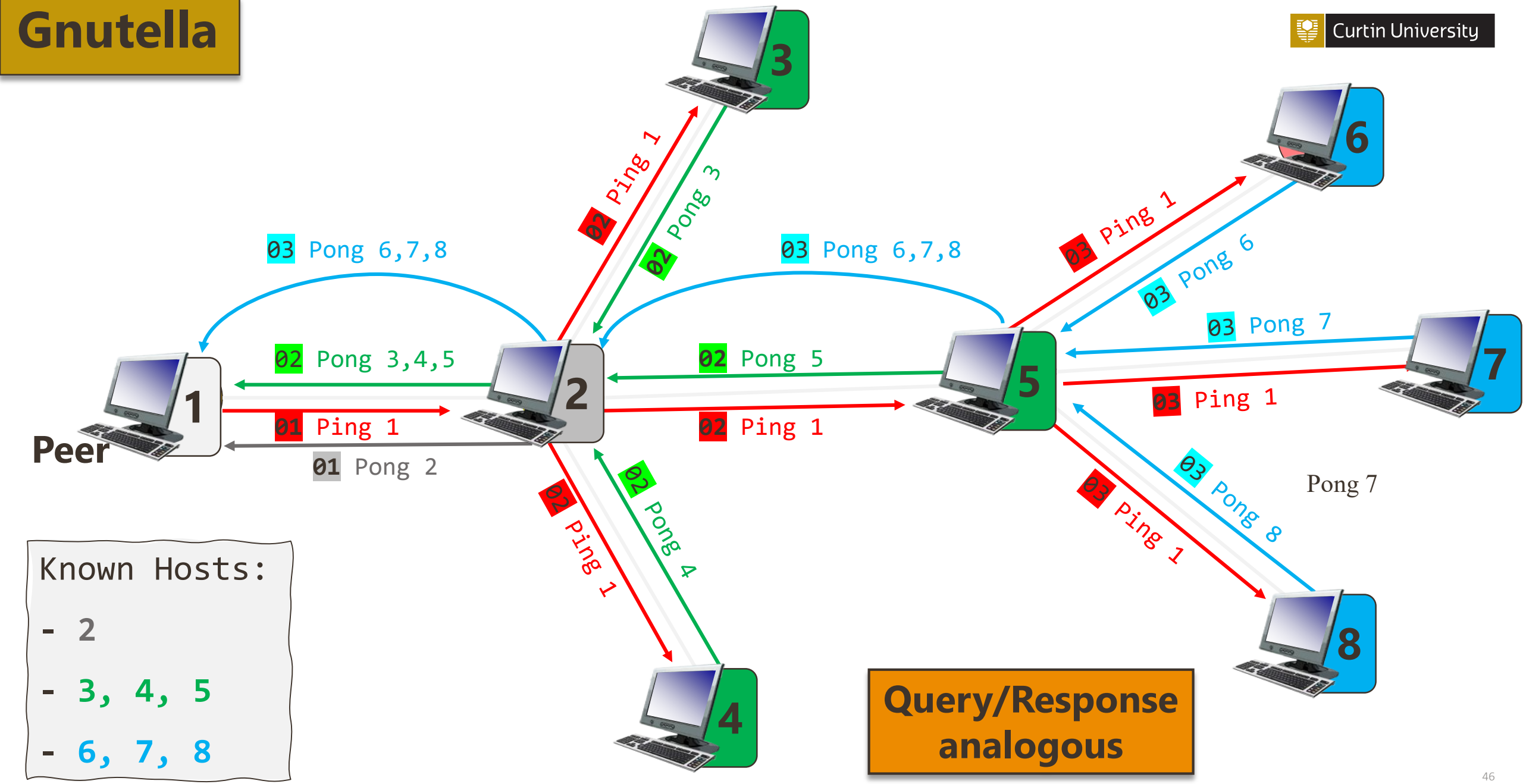
- **Services**
  - Was used primarily for file sharing (audio)
  - Chat program, instant messaging service, tracking program,…
- **Centralized system**
  - Single point of failure => limited fault tolerance
  - Limited scalability (server farms with load balancing)
- **Query is fast** and **upper bound for duration** can be given

Gnutella

Curtin University

- **Pure Peer-To-Peer**
  - ✓ very simple protocol
  - ✓ no routing "intelligence"
- **Constrained Broadcast**
  - ✓ Life-time of packets limited by TTL (typically set to 7)
  - ✓ Packets have unique ids to detect loops

Gnutella


Curtin University
Peer
1
2
3
4
5
6
7
8
01 Ping 1
01 Pong 2
02 Ping 1
02 Pong 3,4,5
02 Pong 3
02 Ping 1
02 Pong 4
02 Ping 1
02 Pong 5
03 Pong 6,7,8
03 Pong 6,7,8
03 Ping 1
03 Pong 6
03 Ping 1
03 Pong 7
03 Ping 1
03 Pong 8
Pong 7
Known Hosts:
- 2
- 3, 4, 5
- 6, 7, 8
Query/Response analogous

Problem - **Free riding**

Curtin University

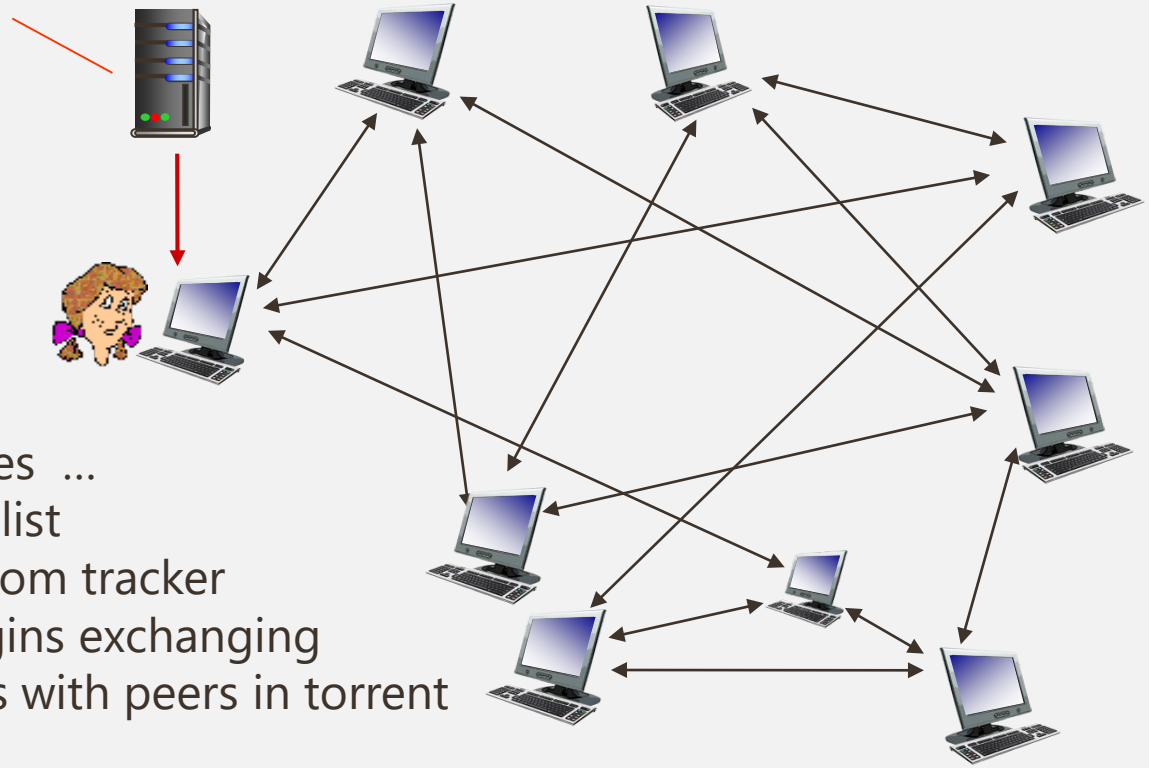
- File sharing networks rely on users sharing data
- Two types of free riding
  1. Downloading but not sharing any data
  2. Not sharing any interesting data
- On Gnutella
  - 15% of users contribute 94% of content
  - 63% of users never responded to a query

BitTorrent: P2P File Distribution

Curtin University

- **File** divided **into** 256Kb **chunks**
- Peers in torrent **send/receive file chunks**

tracker: tracks peers participating in torrent

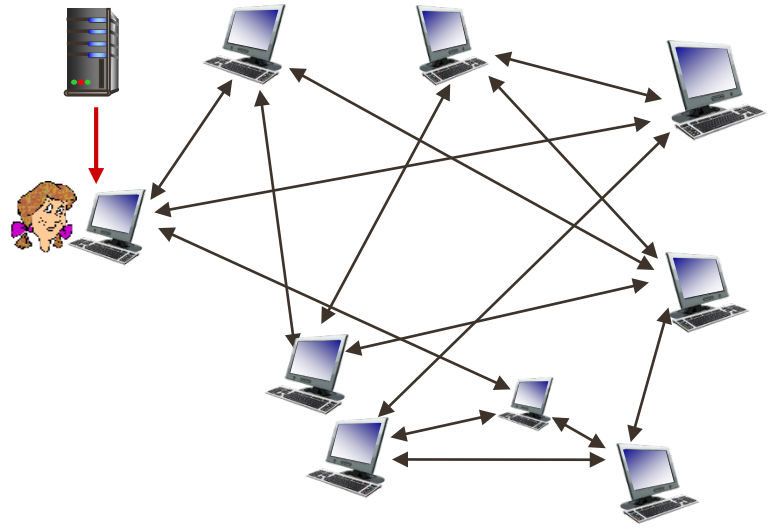


Alice arrives …
… obtains list
of peers from tracker
… and begins exchanging
file chunks with peers in torrent

torrent: group of peers exchanging chunks of a file

BitTorrent – cont.

Curtin University



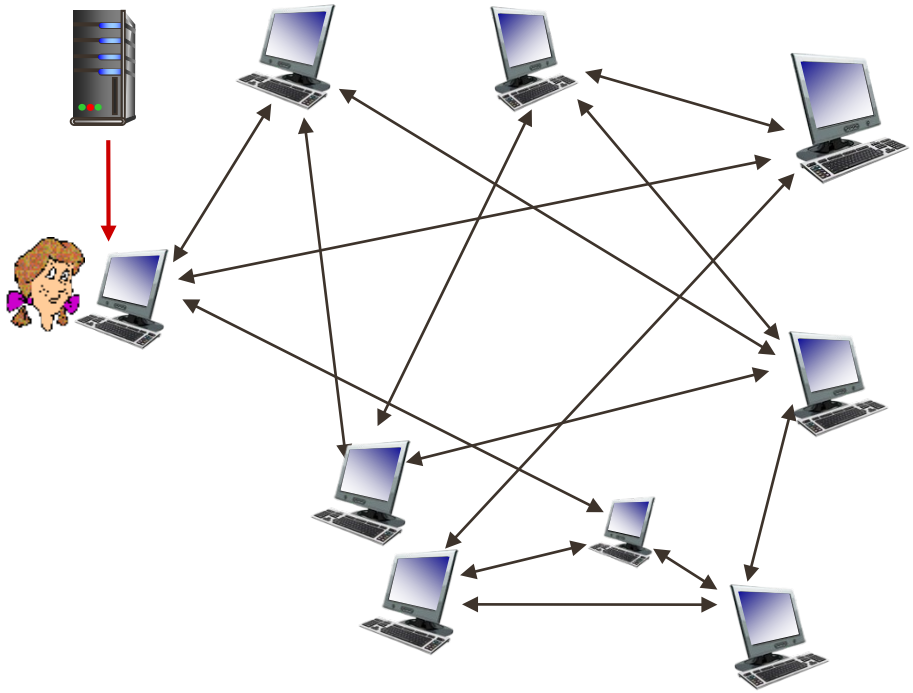
- **Peer joining torrent:**
  - has no chunks, but will accumulate them over time from other peers
  - registers with tracker to get list of peers, connects to subset of peers ("neighbors")

- While downloading, peer uploads **chunks** to other peers
- Peer may change peers with whom it exchanges chunks
- ***churn:*** peers may come and go
- once peer has entire file, it may (selfishly) leave or (altruistically) remain in torrent

Requesting File Chunks

Curtin University

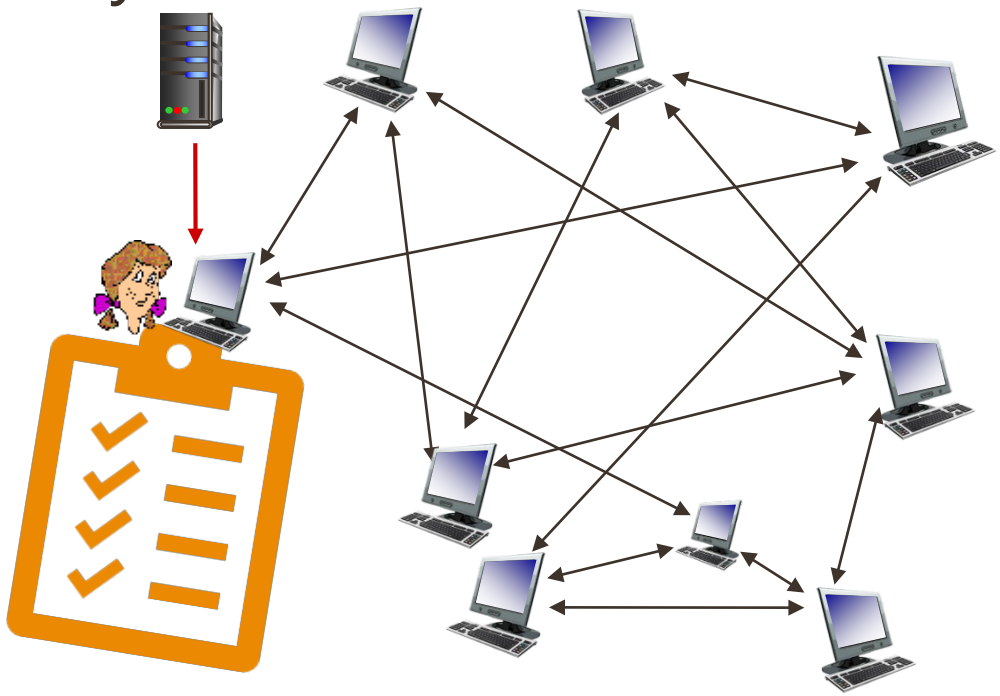
- at any given time, different peers have different subsets of file chunks
- periodically, Alice asks each peer for list of chunks that they have
- Alice requests missing chunks from peers, rarest first



Sending File Chunks (tit-for-tat)

Curtin University

- Alice sends chunks to the **four peers** currently **sending her chunks at highest rate**
  - *Other peers are choked by Alice (do not receive chunks from her)*
  - *Re-evaluate top 4* ***every 10 secs***

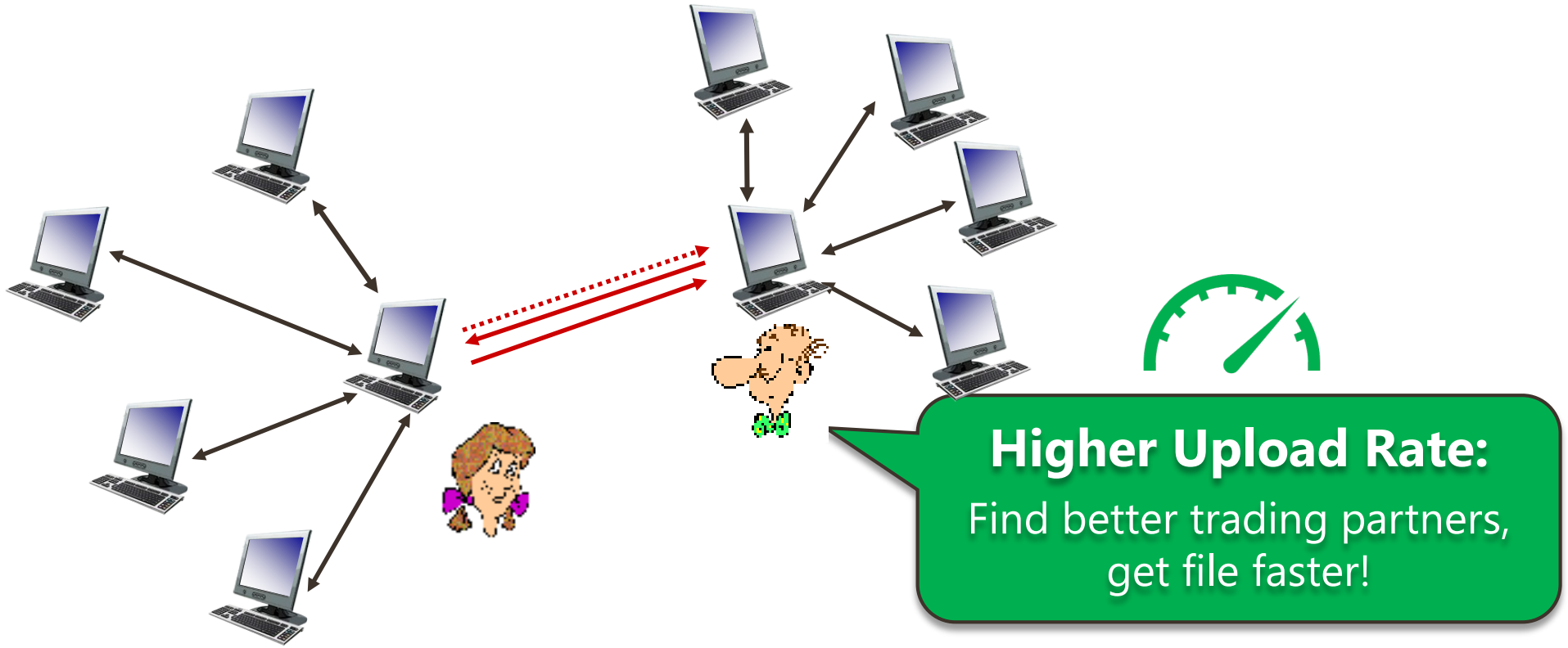


- **Every 30 secs:** randomly select another peer, starts sending chunks
  - **"optimistically unchoke"** this peer
  - newly chosen peer may join top 4

BitTorrent: tit-for-tat

Curtin University

1. Alice **"optimistically unchokes"** Bob
2. Alice becomes one of **Bob's top-four providers**; Bob reciprocates
3. Bob becomes one of **Alice's top-four providers**

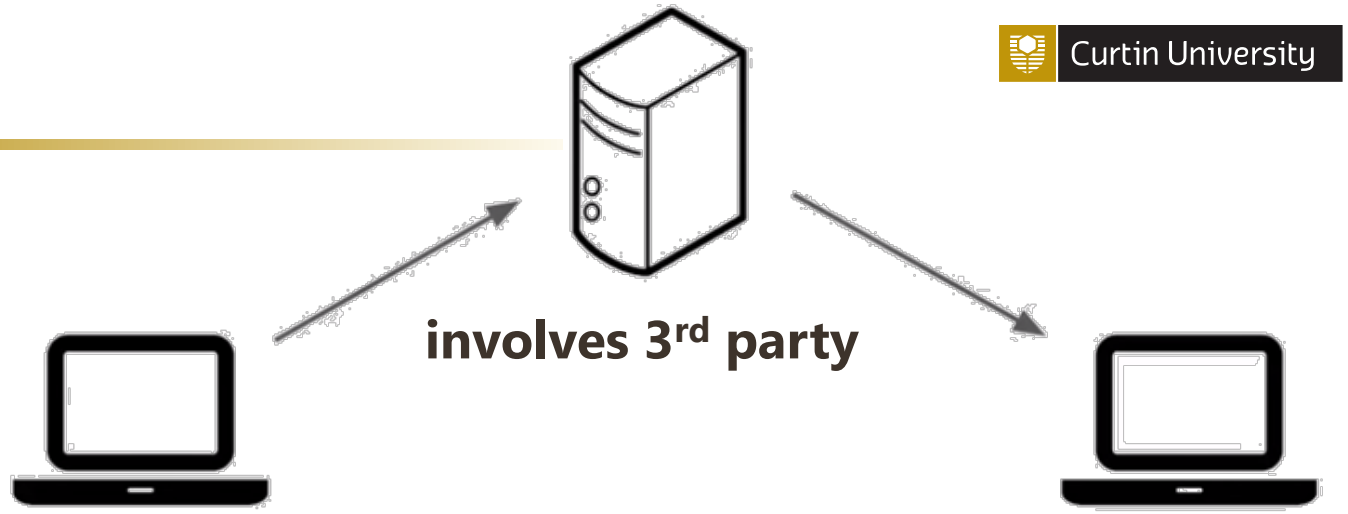


Higher Upload Rate:
Find better trading partners, get file faster!

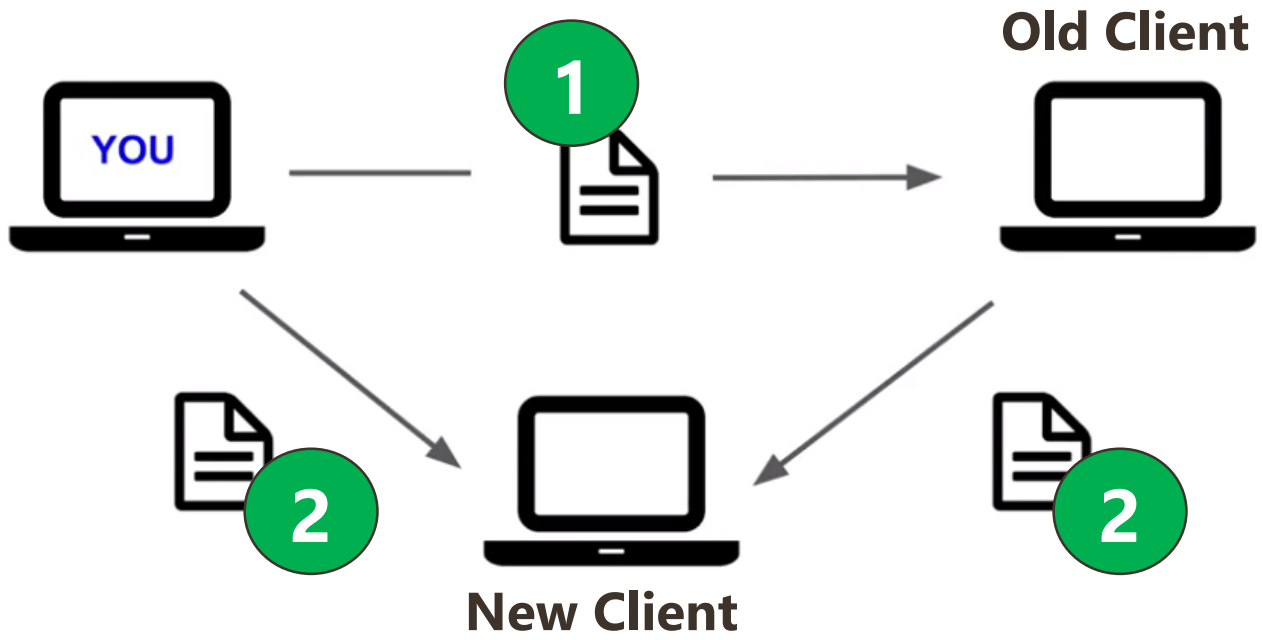
P2P Web

Curtin University

- **Client-Server Model**
  - Lack of privacy, not secure !
    - *Communication is not really end-to-end*
  - Lack of data ownership



- **P2P Web**
  - Improve privacy
  - Give users ownership
  - Build on top of BitTorrent Protocol
  - Files distributed P2P
  - **Visiting users contribute bandwidth**
  - Anyone can publish from their device (can become a server ad-hoc fashion)



Curtin University

Digital Encryption

- Fundamentals
- Symmetric Key Encryption
- Asymmetric Key Encryption
- Digital Signature
- SSL Certificate

What is Encryption?

Curtin University

- Encryption is a process that **encodes a message** or file so that it can be only be read by certain people.

- Encryption uses an **algorithm to scramble, or encrypt, data** and then **uses a key for the receiving party to unscramble**, or decrypt, the information

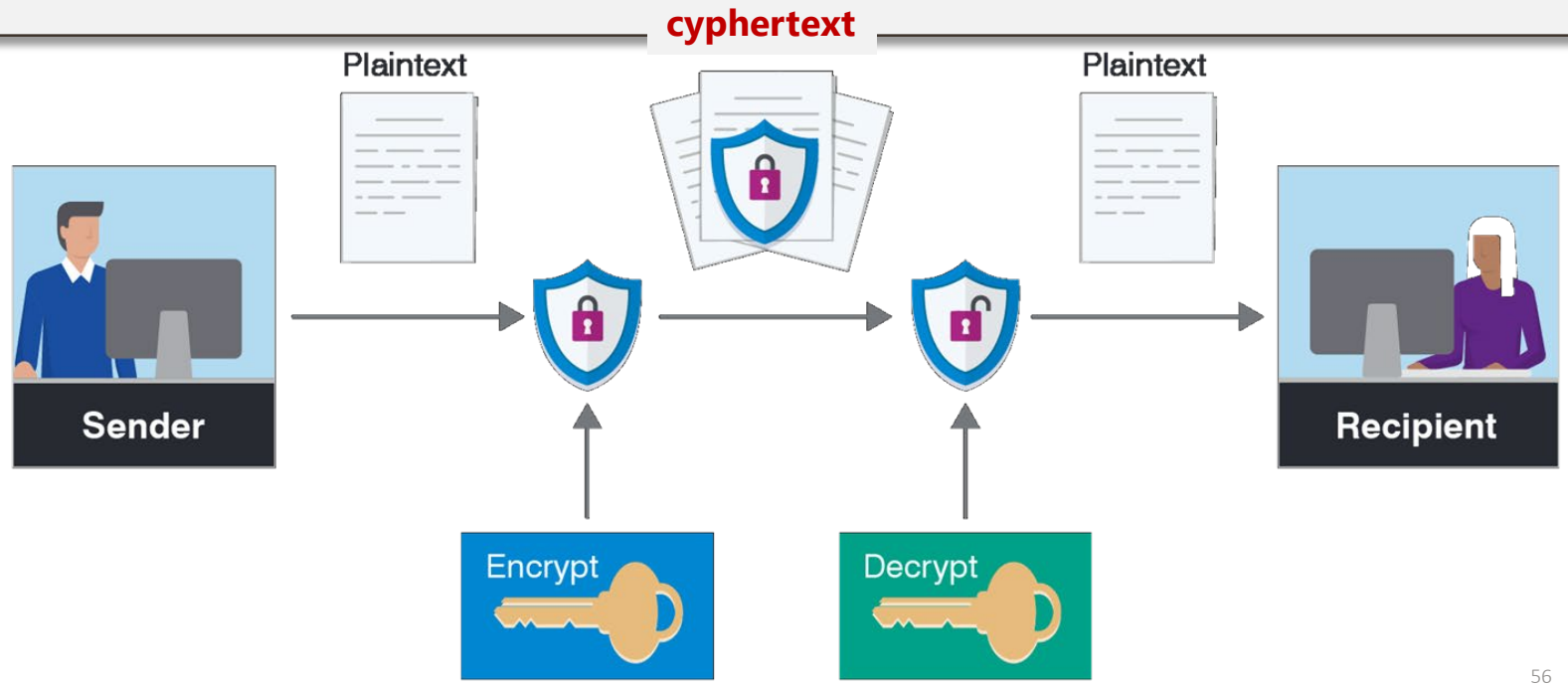
- Types:
  - **Symmetric Key Encryption** (Single Key)
  - **Asymmetric Key Encryption** (Public/Private Key)

Symmetric Key Encryption

Curtin University

- **Only one key** (a secret key) **is used** to both encrypt and decrypt information

- **Encrypt** (Message, **SymKey**) = **Cyphertext**
- **Decrypt** (**Cyphertext**, **SymKey**) = Message



Asymmetric Key Encryption

Curtin University

- Each person has a **key-pair (public key - pk, private key - sk)**

Kept at certification authorities
Some integrated to browsers

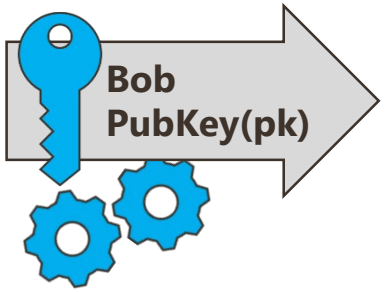
Kept at person (secret)

1. Secrecy:
- **pk** encrypts **sk** decrypt

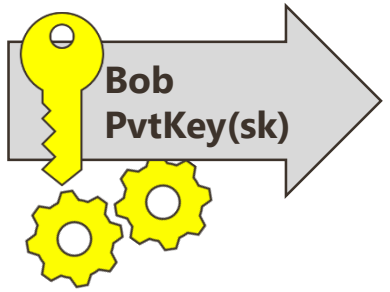
2. Authenticity:
- **sk** can encrypt **pk** decrypt

message

Easter Bunny is Real.



^43&(&)#^@sd a342356asd32ey h21&8)#2139%9 0*(0348kshe48^



message

Easter Bunny is Real.

Digital Signature

Curtin University

- **Same as Handwritten signatures, but digital**
  - To ensure the authenticity of the sender (i.e. a document signed by the sender)

- **Asymmetric Key Encryption**
  - ✓ Any key can be used to encrypt or decrypt
  - ✓ **This make Digital Signatures possible**
- **Hashing**
  - ✓ Generate a hash value
  - ✓ 1-way process (not reversable)
  - ✓ i.e. SHA-128, SHA-256

Digital Signature

Curtin University

Bob's Digital Certificate

message

I agree to pay **Jil** $100

SHA-256

b2148741532594
ahsckoeas321310
345345asfe13905
ghe4825783ferwt

hash value (digest)

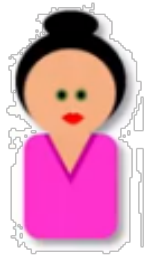
Bob PvtKey(sk)

I agree to pay **Jil** $100

^43&(&)#^@sd
a342356asd32ey
h21&8)#2139%9
0*(0348kshe48^

Encrypted hash value

Sign(Message, sk_{bob}) = Signature



message

I agree to pay Jil $100

SHA-256

b2148741532594
ahsckoeas321310
345345asfe13905
ghe4825783ferwt

?

^43&(&)#^@sd
a342356asd32ey
h21&8)#2139%9
0*(0348kshe48^

Bob PubKey(pk)

b2148741532594
ahsckoeas321310
345345asfe13905
ghe4825783ferwt

Encrypted hash value

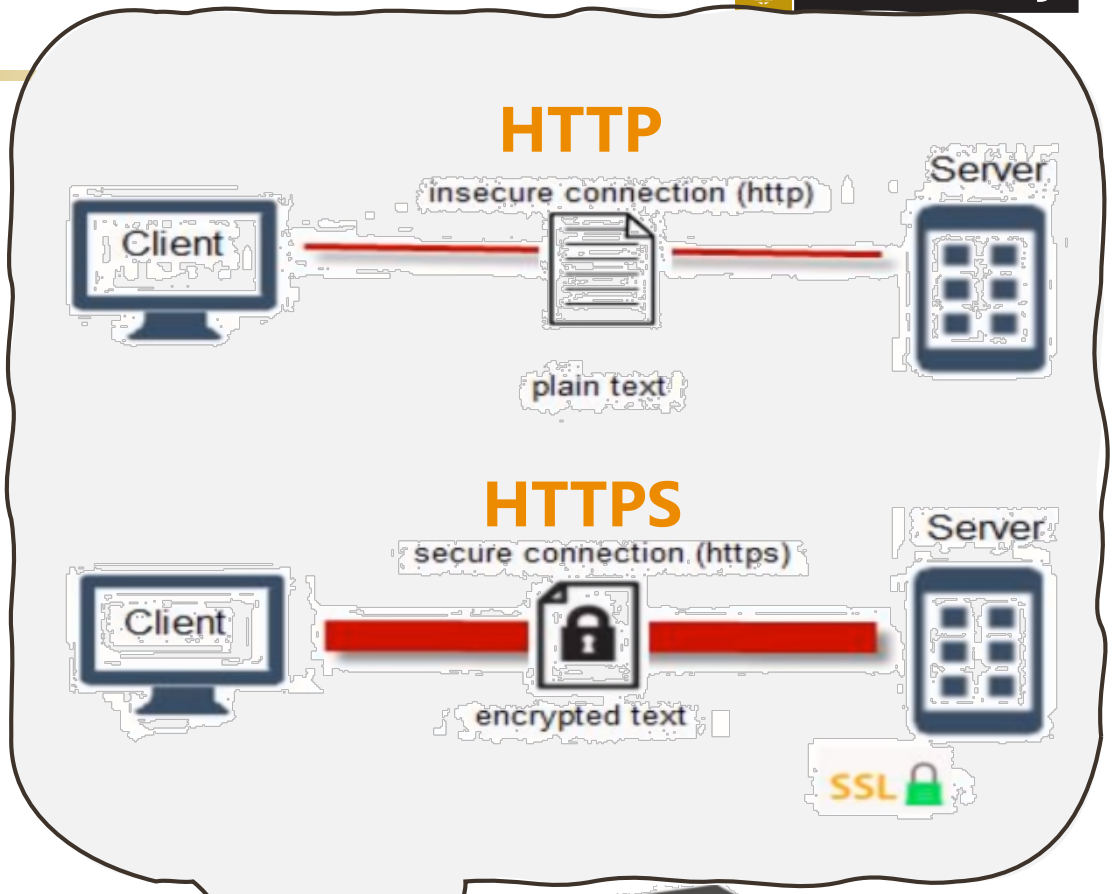
Verify(Message, Signature, pk_{bob}) = T/F

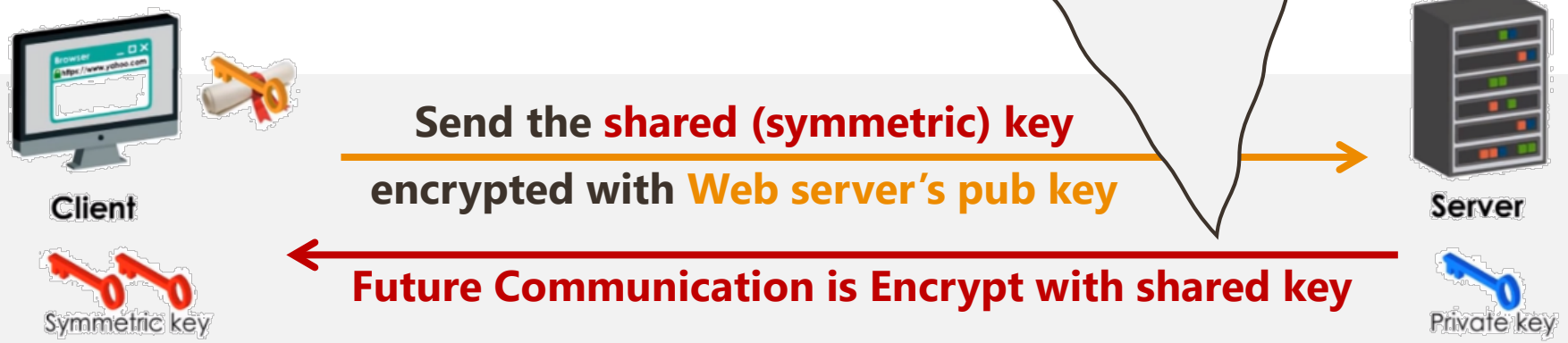
SSL Certificate

Curtin University

Web Server's Digital Certificate

- Verified Identity with the Public Key





Curtin University

- **App Protocols – DHCP**
  - 4-way Handshake
    - DHCP Discover
    - DHCP Offer
    - DHCP Request
    - DHCP Ack
- **Domain Name System (DNS)**
  - hosts File
  - Main Elements
    - Domain Name Space
    - Name Server
    - Resolvers
  - DNS Database
  - DNS Registrar
- **Web Search Engines**
  - Fundamentals
  - Ranking Algorithm
  - Search Results
- **Web Services**
  - REST API
  - URI / URL / URN
  - REST API Highlights
- **P2P**
  - Fundamentals
  - Napster
  - Gnutella
  - BitTorrent – *in depth*
  - P2P Web
- **Digital Encryption**
  - Fundamentals
  - Symmetric Key Encryption
  - Asymmetric Key Encryption
  - Digital Signature
  - SSL Certificate


Curtin University
314
THANK YOU
Make tomorrow better.