

# Planificación y Administración de Redes: Domain Name System



IES Gonzalo Nazareno  
**CONSEJERÍA DE EDUCACIÓN**

Jesús Moreno León  
Raúl Ruiz Padilla  
j.moreno1@gmail.com  
Septiembre 2010

---

Estas diapositivas son una obra derivada de las transparencias  
del Grupo de Sistemas y Comunicaciones  
de la Universidad Rey Juan Carlos  
Puede encontrarse una versión de este documento en  
<http://gsyc.es/moodle>

© Jesús Moreno León, Raúl Ruiz Padilla, Septiembre de 2010

Algunos derechos reservados.  
Este artículo se distribuye bajo la licencia  
"Reconocimiento-CompartirIgual 3.0 España" de Creative  
Commons, disponible en  
<http://creativecommons.org/licenses/by-sa/3.0/es/deed.es>

Este documento (o uno muy similar)  
está disponible en (o enlazado desde)  
<http://informatica.gonzalonazareno.org>

# ¿Por qué necesitamos el DNS?

---

Los humanos preferimos nombres a direcciones IP

- nba.com frente a 80.67.85.152

Los números IP están ligados a la estructura de la red

- google.com y google.es tendrán IPs muy diferentes

Los números IP están ligados a máquinas concreta

- nba.com puede cambiar de máquina y de IP, pero no de nombre

Es necesario establecer una correspondencia entre nombres y direcciones IP



## Un poco de historia

---

Al principio (años 70), existía un único fichero (HOSTS.TXT) con información de todas las máquinas de ARPANET

Se obtenía periódicamente por FTP. Gestión completamente manual

Con el crecimiento de ARPANET se hizo inmanejable

1984, primeras RFC que describen un servicio de nombres basado en dominios

En 1987, **RFC 1034** y **RFC 1035** que describen DNS



# Estructura de nombrado

---

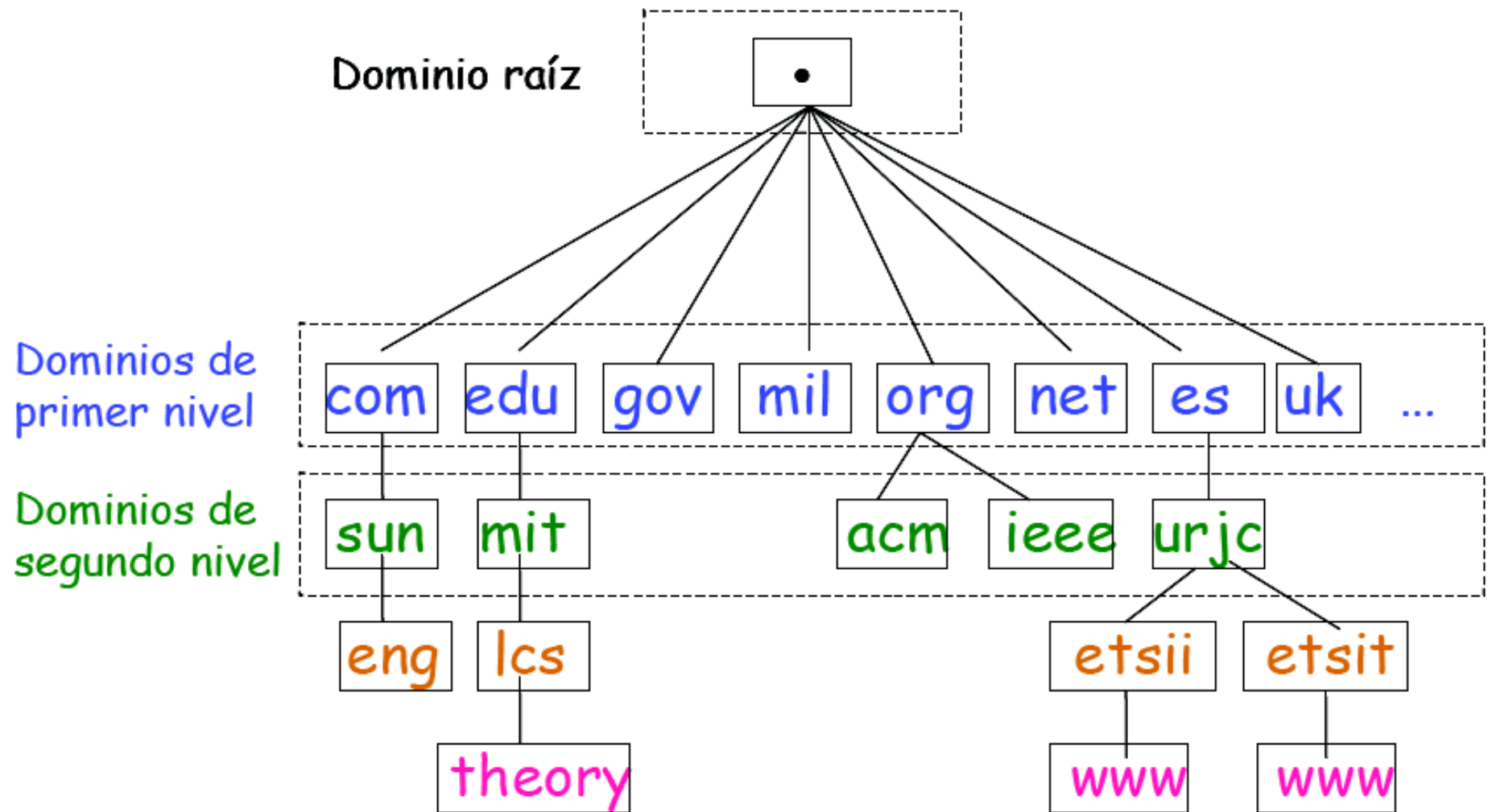
Se descentraliza el control consiguiendo una estructura jerárquica y fácilmente ampliable

Jerarquía de dominios:

- Dominio raíz (root o “.”). Gestionado por ICANN
- Dominios de primer nivel (TLDs, Top Level Domains)  
com, edu, gov, mil, net, org, int, info, museum,  
jobs,... y códigos ISO de países (uk, mx, ar, de,  
es. . . )
- Dominios de segundo nivel, tercer nivel...



# Arbol de dominios



# Dominios, subdominios y zonas

---

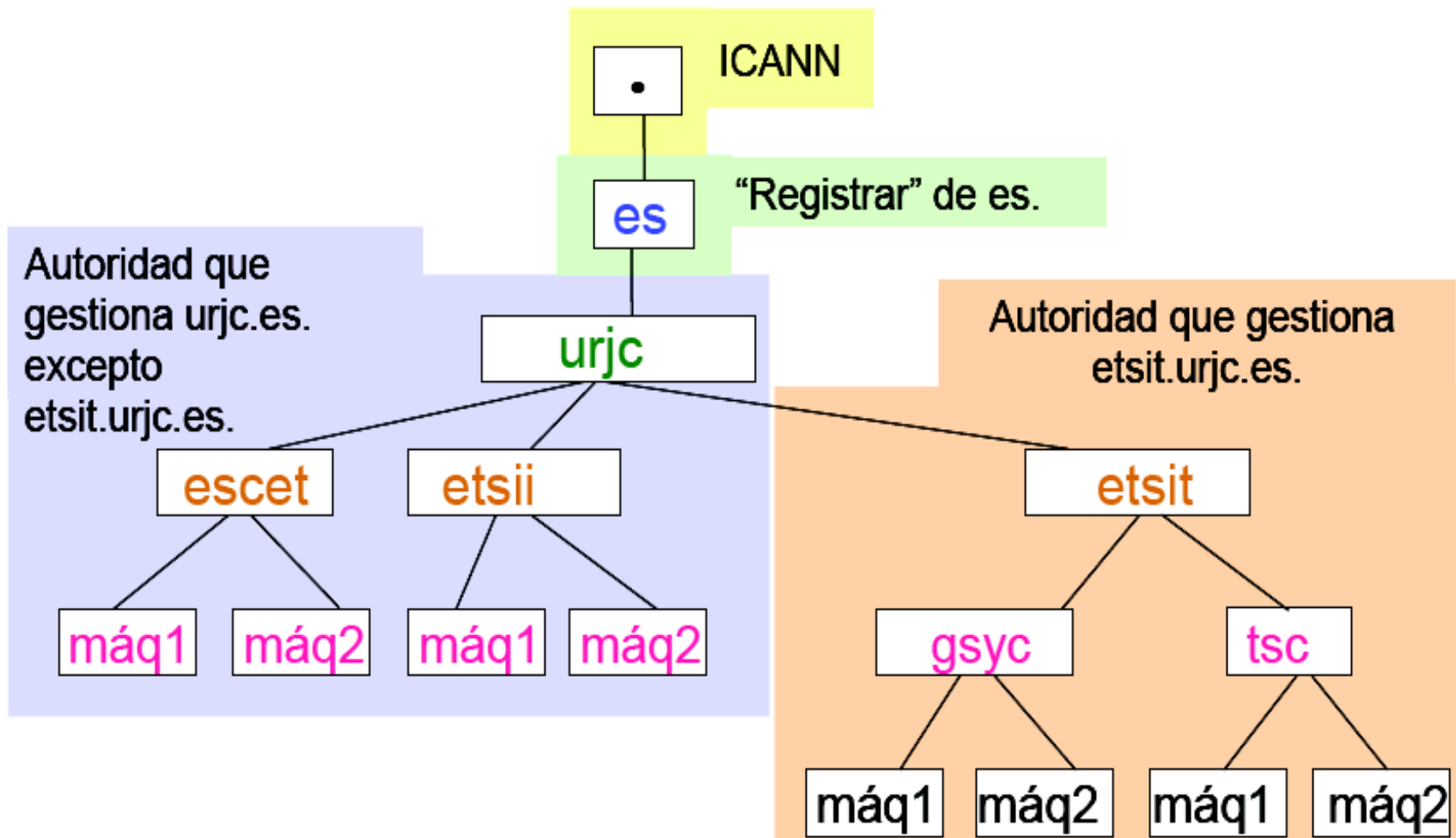
La organización o autoridad encargada de la administración de un dominio puede decidir dividirlo en subdominios

Los subdominios pueden seguir administrados por la misma autoridad, o puede delegarse la responsabilidad de su administración a otras organizaciones

Se llama zona a un subárbol de DNS administrado por una organización diferente a la que administra su dominio padre



# Dominios, subdominios y zonas





# Dominio directo y dominio inverso

---

- El dominio directo proporciona para cada nombre una dirección IP
- El dominio inverso proporciona para cada dirección IP un nombre

También se conoce como dominio in-addr.arpa.

La red 172.16.0.0 es el dominio inverso  
16.172.in-addr.arpa



# Esquema de funcionamiento

---

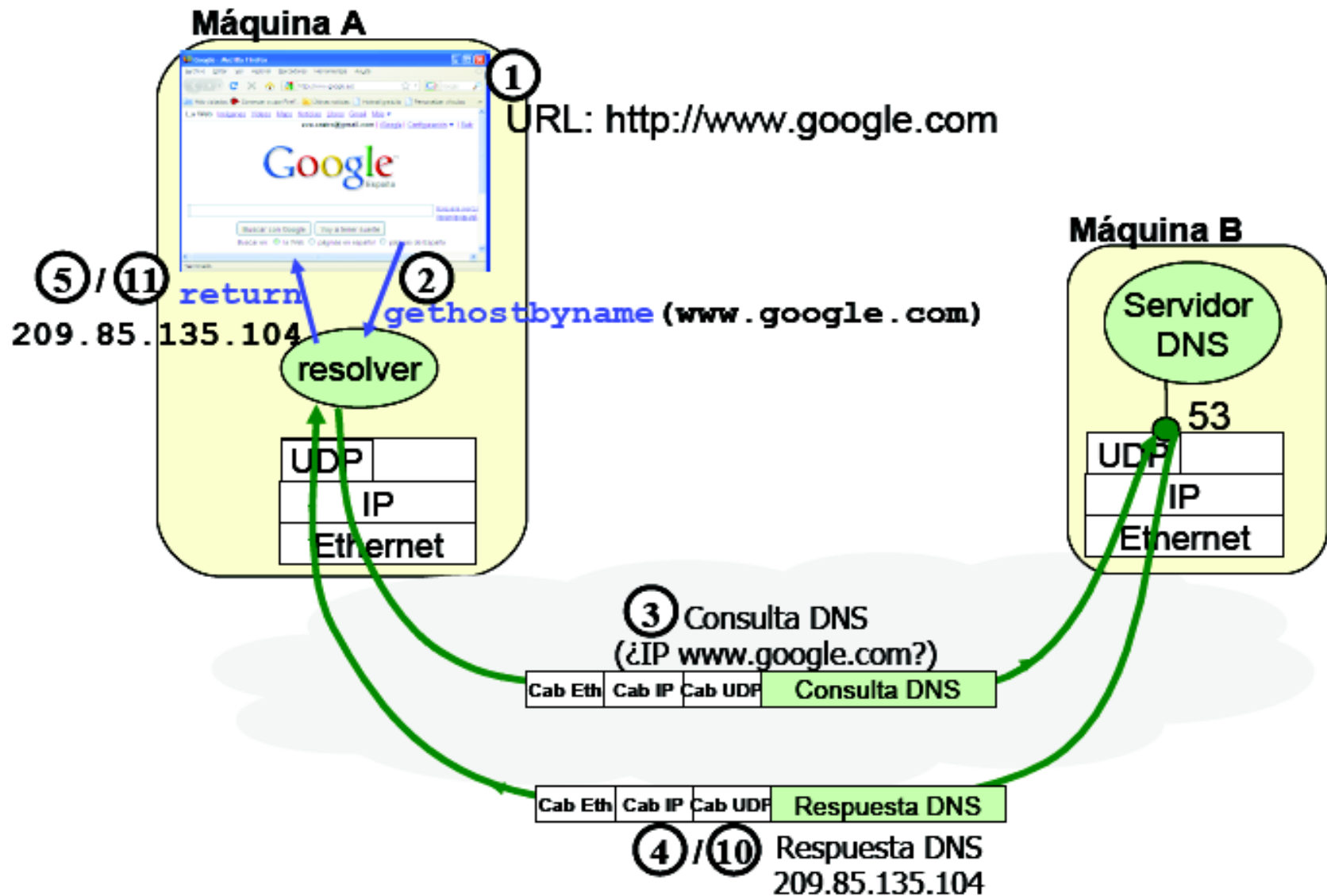
Se trata de mantener la información como una Base de Datos Distribuida. Las consultas al DNS se realizan en modo cliente-servidor:

Cuando una aplicación (cliente) quiere resolver un nombre (que no aparece en el fichero local), preguntará a un servidor DNS a través una biblioteca de consulta al DNS llamada *resolver*

El servidor DNS investiga por su cuenta (si no sabe la IP de la máquina) y devuelve la IP pedida



# Consulta a un servidor



# Resolución de nombres en un servidor DNS

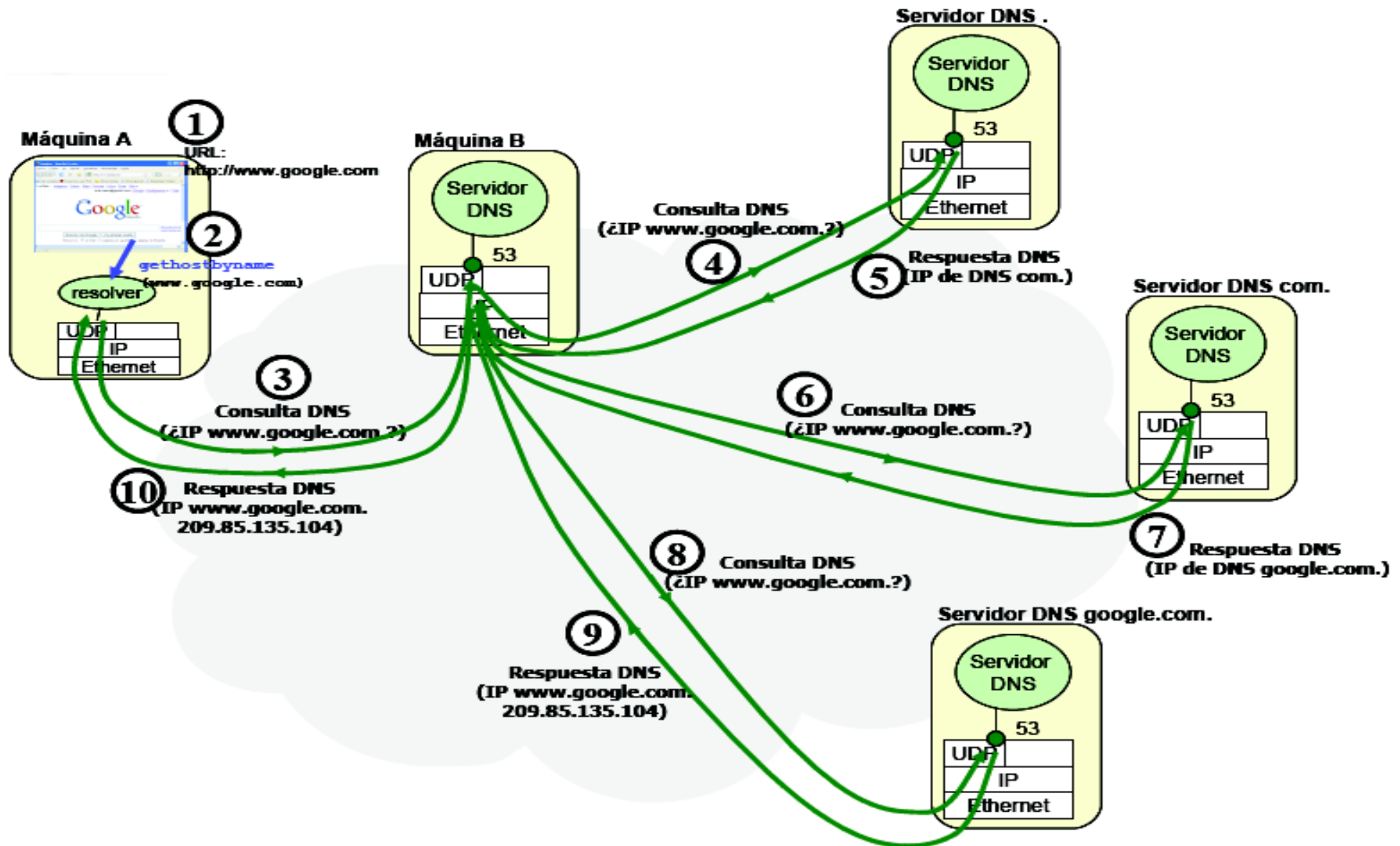
---

Cuando un servidor recibe una consulta para resolver un nombre (por ejemplo `www.google.com`):

1. Comprueba si el nombre pertenece a alguno de los dominios que sirve. Si es así, busca en su “mapa” y devuelve la IP correspondiente
2. Si no, pregunta a un servidor del dominio raíz, que le contestará con la IP de un servidor del dominio com
3. Luego pregunta a ése, obteniendo la IP de un servidor del dominio `google.com`
4. Ahora se pregunta al último, que ya tiene en su mapa la IP buscada



# Resolución de nombres en un servidor DNS



# Cachés en el servicio DNS

---

Cuando un servidor, al realizar una búsqueda, aprende un dato que no sabía, lo guarda en una caché. Si vuelve a necesitar ese dato, lo saca de la caché en vez de volver a preguntar.

Los mapas de dominio especifican el tiempo que puede estar ese dato en una caché (ttl)

En algunos SO el resolver mantiene su propia caché con los resultados de búsquedas previas y algunas aplicaciones también lo hacen.



# Algo de nomenclatura

---

## Consultas a un servidor

- Recursivas
- Iterativas

## Tipos de servidores

- Primario (maestro)
- Secundario (esclavo) – transferencias de zona
- Reenviadores (forwarders)

## Respuesta con autoridad (authoritative)





# Root nameservers





# Mapas de dominio

Nombre	TTL	Clase	Tipo	Valor
\$ORIGIN	gsyc.es.			; añadido a los nombres no terminados en punto
\$TTL	86400			; ttl por defecto: 1 día. También: 1d o 24h
gsyc.es.		IN	SOA	ns1.gsync.es. admin-gsync.gmail.com. ( 2008030201 ; N° de serie 8h ; Refresco 2h ; Reintento 7d ; Expiración 1d ) ; ttl para respuestas negativas ; (de nombre inexistente)
gsync.es.		IN	NS	ns1.gsync.es.
gsync.es.		IN	NS	ns2.gsync.es.
gsync.es.	7200	IN	MX	correo.gsync.es.
ns1.gsync.es.		IN	A	193.147.71.5
ns2.gsync.es.		IN	A	193.147.71.6
tierra.gsync.es.		IN	A	193.147.71.7
hielo.gsync.es.		IN	A	193.147.71.8
agua.gsync.es.		IN	A	193.147.71.9
fuego.gsync.es.		IN	A	193.147.71.10
www.gsync.es.	14400	IN	CNAME	agua.gsync.es.
mail.gsync.es.	14400	IN	CNAME	fuego.gsync.es.
aulas.gsync.es.		IN	NS	ns.aulas.gsync.es.
ns.aulas.gsync.es.		IN	A	212.135.11.45

# Formato de los mensajes DNS

