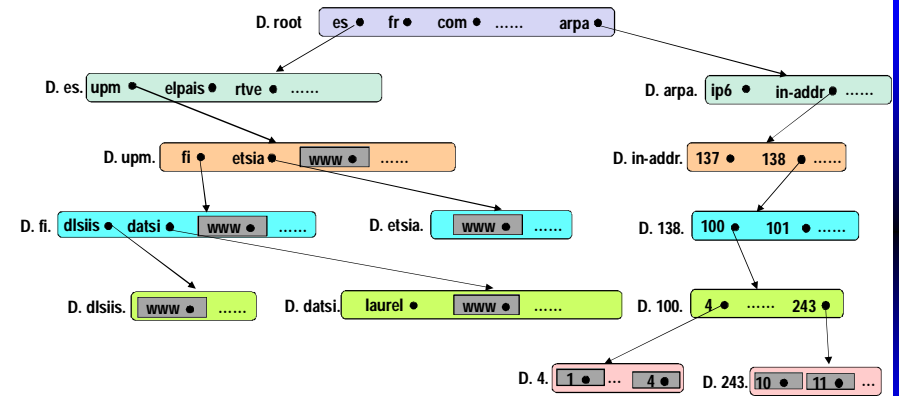


Domain Name System (DNS)

- Servicio de nombres de máquinas en Internet: nombre → IP
 - No es un serv. nombres general pero ilustrativo por escalabilidad
 - Diseño genérico: aunque uso habitual nombre de máquinas Internet
 - Inicios de Internet: fichero HOST que se actualizaba periódicamente
- Espacio de nombres de DNS jerárquico
 - Nombre: secuencia de dominios (≈directorios) de dcha. a izda.
 - www.datsi.fi.upm.es. → . + es + upm + fi + datsi
 - Dominio raíz: . → Caminos absolutos (FQDN) terminan con .
 - Dominios nivel superior (TLD)
 - gTLDs: genéricos (com, org, ...)
 - ccTLDs: por país (¿qué pasa con el de Tuvalu?)
 - De segundo nivel, de tercero, ...
- Implementación más usada BIND

Jerarquía de dominios DNS



Espacio de nombres distribuido: Zonas

- Zona DNS: partición del árbol global (zona ≠ dominio)
 - Información recursos de un dominio y sus subdominios no delegados
 - Delegación de dominios
 - Un subdominio puede tener su propia zona
 - Dominio padre incluye "punto de montaje" a esa zona subordinada
 - Diseño habitual: delegar todos los subdominios
 - Una zona para cada dominio (zona ≈ dominio)
 - Incluso a veces usando el mismo servidor primario que el dominio padre
 - Dominio y subdominio con mismo administrador
- Cada zona está replicada (comportamiento PAEL):
 - 1 servidor maestro/primario y N (al menos 1) esclavos/secundarios
 - Fiabilidad: mejor réplicas en distintas subredes
- Información contenida en una zona:
 - Colección de *Resource Records* (RR) que describen sus recursos

Resource Record

- Definición de un recurso: *Nombre TTL Clase Tipo Datos*
 - Clase *IN* para Internet (otros *HS*, para Hesiod, y *CH*, para Chaos)

www.fi.upm.es.	86400	IN	A	138.100.243.10
----------------	-------	----	---	----------------
- Fichero de zona:
 - Fichero de texto en serv. primario define RRs de una zona: 1 RR/línea
 - Aunque RRs se transmiten en binario
 - Incluye RRs de recursos del dominio y de subdominios no delegados
 - Sintaxis definida para facilitar introducción de datos en fichero de zona
 - Macros, caracteres especiales, caminos relativos, omisión de campos,...
- Diversos tipos de RRs
 - Nos centramos en SOA, A, AAAA, PTR, CNAME, MX, SRV, TXT y NS
 - No tratamos los RRs relacionados con la extensión DNSSEC
 - Proporciona autenticación e integridad en DNS

Ejemplo fichero de zona (Wikipedia)

```
example.com. 3600 IN SOA ns.example.com. username.example.com. ( 2007120710 86400 7200 2419200 3600 )
example.com. 3600 IN NS ns.example.com. ; ns.example.com is a nameserver for example.com
example.com. 3600 IN NS ns.somewhere.example. ; backup nameserver for example.com
example.com. 3600 IN MX 10 mail.example.com. ; mail.example.com is the mailserver for example.com
example.com. 3600 IN MX 20 mail2.example.com. ;
example.com. 3600 IN MX 50 mail3.example.com. ;
example.com. 3600 IN A 192.0.2.1 ; IPv4 address for example.com
example.com. 3600 IN AAAA 2001:db8:10::1 ; IPv6 address for example.com
ns.example.com. 3600 IN A 192.0.2.2 ; IPv4 address for ns.example.com
ns.example.com. 3600 IN AAAA 2001:db8:10::2 ; IPv6 address for ns.example.com
www.example.com. 3600 IN CNAME example.com. ; www.example.com is an alias for example.com
wwwtest.example.com. 3600 IN CNAME www.example.com. ; another alias for www.example.com
mail.example.com. 3600 IN A 192.0.2.3 ; IPv4 address for mail.example.com
mail2.example.com. 3600 IN A 192.0.2.4 ; IPv4 address for mail2.example.com
mail3.example.com. 3600 IN A 192.0.2.5 ; IPv4 address for mail3.example.com
```

RR de tipo SOA (Start of Authority)

- Comienzo de definición de una zona
- Ejemplo de definición en fichero de zona (wikipedia)

```
example.com. IN SOA ns.example.com. username.example.com. (
    2007120710 ; serial number of this zone file
    1d ; slave refresh (1 day)
    2h ; slave retry time in case of a problem (2 hours)
    4w ; slave expiration time (4 weeks)
    1h ; maximum caching time in case of failed lookups (1 hour)
)
```

- Ejemplo de consulta: *dig fi.upm.es. SOA*

```
fi.upm.es. 86400 IN SOA chita.fi.upm.es. hostmaster.fi.upm.es. 2013102101 28800 7200 2419200 3600
```

Dominio; TTL; Clase Internet; Start Of Authority; S. maestro; responsable; n° serie (incrementar si cambio);

Periodo de actualización de secundario; Tiempo de reintento de secundario antes actualización fallida;

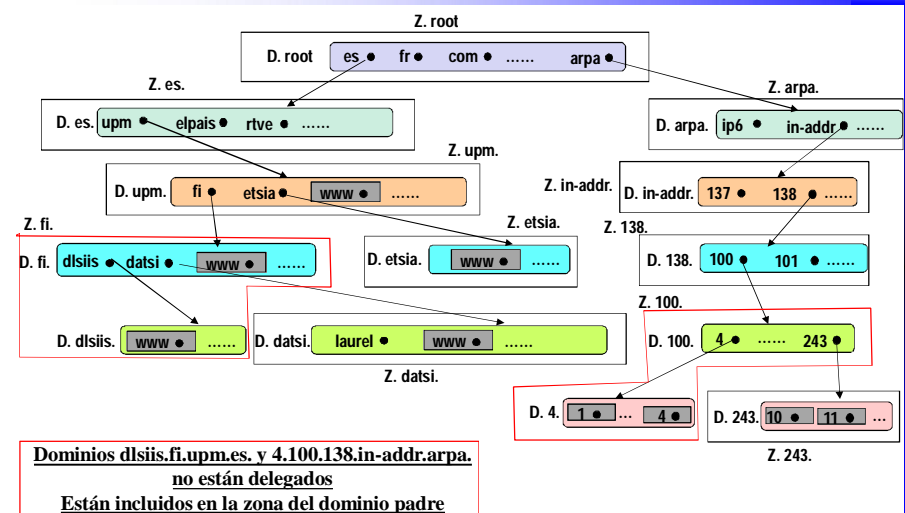
Tiempo de expiración de info. de secundario ante sincronización fallida; TTL para cache negativa (tiempo en cache de consultas erróneas)

- Si dominio está delegado → tiene SOA; tiene su propio fichero de zona

Registros SOA

- *dig . SOA:* . 44659 IN SOA a.root-servers.net. nsld.verisign-grs.com. 2016030901 1800 900 604800 86400
- *dig es. SOA:* es. 44796 IN SOA ns1.nic.es. hostmaster.nic.es. 2016030903 7200 7200 2592000 86400
- *dig upm.es. SOA:* upm.es. 44873 IN SOA einstein.ccupm.upm.es. hostmaster.upm.es. 2016022602 86400 7200 1209600 3600
- *dig etsia.upm.es. SOA:* etsia.upm.es. 45075 IN SOA einstein.ccupm.upm.es. hostmaster.upm.es. 2016022901 86400 7200 1209600 7200
- *dig fi.upm.es. SOA:* fi.upm.es. 86400 IN SOA chita.fi.upm.es. hostmaster.fi.upm.es. 2016021601 28800 7200 2419200 3600
- *dig datsi.fi.upm.es. SOA:* datsi.fi.upm.es. 86400 IN SOA chita.fi.upm.es. hostmaster.fi.upm.es. 2015120901 28800 7200 2592000 3600
- *dig dlsiis.fi.upm.es. SOA:* No hay respuesta
- *dig arpa. SOA:* arpa. 45018 IN SOA a.root-servers.net. nsld.verisign-grs.com. 2016030900 1800 900 604800 86400
- *dig in-addr.arpa. SOA:* in-addr.arpa. 3600 IN SOA b.in-addr-servers.arpa. nsld.iana.org. 2015073098 1800 900 604800 3600
- *dig 138.in-addr.arpa. SOA:* 138.in-addr.arpa. 44938 IN SOA z.arin.net. dns-ops.arin.net. 2016013238 1800 900 691200 10800
- *dig 100.138.in-addr.arpa. SOA:* 100.138.in-addr.arpa. 44773 IN SOA einstein.ccupm.upm.es. hostmaster.upm.es. 2016022901 86400 7200 1209600 7200
- *dig 4.100.138.in-addr.arpa. SOA:* No hay respuesta
- *dig 243.100.138.in-addr.arpa. SOA:* 243.100.138.in-addr.arpa. 86400 IN SOA chita.fi.upm.es. hostmaster.fi.upm.es. 2014091801 28800 7200 2592000 3600

Jerarquía de zonas DNS



Zonas y servidores primarios

- zona . a.root-servers.net
- zona .es ns1.nic.es
- zona upm.es. einstein.ccupm.upm.es
- zona etsia.upm.es. einstein.ccupm.upm.es.
 - Mismo servidor primario que su padre → mismo administrador
- zona fi.upm.es. chita.fi.upm.es.
 - Incluye subdominio dlsi.fi.upm.es.
- zona datsi.fi.upm.es. chita.fi.upm.es.
 - Mismo servidor primario que su padre → mismo administrador
- zona arpa. a.root-servers.net.
 - Mismo servidor primario que su padre → mismo administrador
- zona in-addr.arpa. b.in-addr-servers.arpa.
- zona 138.in-addr.arpa. z.arin.net.
- zona 100.138.in-addr.arpa. einstein.ccupm.upm.es.
 - Incluye subdominio 4.100.138.in-addr.arpa.
- zona 243.100.138.in-addr.arpa. chita.fi.upm.es.

RR de tipo A o AAAA

- Dirección de máquina: A (IPv4) y AAAA (IPv6)

- Ejemplo: *dig www.fi.upm.es. A*

```
www.fi.upm.es. 86400 IN A 138.100.243.10
```

- Ejs: *dig galileo.ccupm.upm.es. A*

```
galileo.ccupm.upm.es. 76103 IN A 138.100.4.4
```

- Ej: *dig galileo.ccupm.upm.es. AAAA*

```
galileo.ccupm.upm.es. 76223 IN AAAA 2001:720:41c:40:12:100:4:4
```

- Múltiples recursos con mismo nombre (reparto de carga)

```
yahoo.es. 300 IN A 98.137.236.24
```

```
yahoo.es. 300 IN A 77.238.184.24
```

```
yahoo.es. 300 IN A 74.6.50.24
```

```
yahoo.es. 300 IN A 212.82.102.24
```

```
yahoo.es. 300 IN A 106.10.212.24
```

- Nótese TTL bajo en RR para favorecer el reparto de carga

RR de tipo PTR

- Traducción inversa dirección IP → Nombre
- Gestionada también por DNS: mediante dominios especiales
- Para IPV4: *in-addr.arpa*.
 - Traducir 138.100.243.10 → 10.243.100.138.in-addr.arpa.
- Para IPV6: *ip6.arpa*.
 - Traducir 2001:720:41c:40:12:100:4:4 → 4.0.0.0.4.0.0.0.0.1.0.2.1.0.0.0.4.0.0.c.1.4.0.0.2.7.0.1.0.0.2.ip6.arpa.
- Ejemplos de consulta:
 - *dig 10.243.100.138.in-addr.arpa. PTR*

```
10.243.100.138.in-addr.arpa. 86400 IN PTR www.fi.upm.es.
```

- *dig 4.0.0.0.4.0.0.0.0.1.0.2.1.0.0.0.4.0.0.c.1.4.0.0.2.7.0.1.0.0.2.ip6.arpa. PTR*

```
4.0.0.0.4.0.0.0.0.1.0.2.1.0.0.0.4.0.0.c.1.4.0.0.2.7.0.1.0.0.2.ip6.arpa. 86302 IN PTR galileo.ccupm.upm.es.
```

RR de tipo CNAME (Canonical NAME)

- Alias: Nuevo nombre para mismo recurso (ejemplo real)

```
www.datsi.fi.upm.es. 86400 IN CNAME avellano.datsi.fi.upm.es.
```

```
avellano.datsi.fi.upm.es. 86400 IN A 138.100.9.22
```

- Frente a (ejemplo hipotético):

```
www.datsi.fi.upm.es. 86400 IN A 138.100.9.22
```

```
avellano.datsi.fi.upm.es. 86400 IN A 138.100.9.22
```

- Más flexibilidad ante cambios pero ineficiencia por indirección

- Pueden encadenarse:

```
www.elpais.com. 967 IN CNAME elpais.es.edgesuite.net.
```

```
elpais.es.edgesuite.net. 10467 IN CNAME a1749.g.akamai.net.
```

```
a1749.g.akamai.net. 20 IN A 130.206.192.24
```

```
a1749.g.akamai.net. 20 IN A 130.206.192.49
```

RR de tipo MX

- Servidores de correo para dominio con orden de preferencia
- Formato: *Nombre TTL clase MX prioridad servidor*
- Ejemplo de consulta: *dig upm.es. MX*

```
upm.es.      73788 IN    MX    10 relay.upm.es.  
upm.es.      73788 IN    MX    30 relay4.upm.es.  
upm.es.      73788 IN    MX    50 correo.upm.es.
```

- Ejemplo de consulta: *dig fi.upm.es. MX*

```
fi.upm.es.   60 IN    MX    10 relay.fi.upm.es.  
fi.upm.es.   60 IN    MX    100 relay.upm.es.  
fi.upm.es.   60 IN    MX    600 relay.fi.upm.es.
```

- Número indica orden de preferencia: ↓ prioridad → ↑ preferencia
- Remitente de correo debe contactar con servidor de menor n°
 - Si caído con el siguiente, ...

RR de tipo SRV

- Permite especificar qué máquinas dan un servicio en el dominio
- Formato: *_servicio._protocolo.nombre TTL clase SRV prioridad peso puerto servidor*
- Permite especificar prioridades y reparto entre misma prioridad
- Ejemplo de wikipedia:

```
_sip._tcp.example.com. 86400 IN SRV 10 60 5060 bigbox.example.com.  
_sip._tcp.example.com. 86400 IN SRV 10 20 5060 smallbox1.example.com.  
_sip._tcp.example.com. 86400 IN SRV 10 10 5060 smallbox2.example.com.  
_sip._tcp.example.com. 86400 IN SRV 10 10 5066 smallbox2.example.com.  
_sip._tcp.example.com. 86400 IN SRV 20 0 5060 backupbox.example.com.
```

- Ejemplo real: *dig SRV _xmpp-server._tcp.gmail.com.*

```
_xmpp-server._tcp.gmail.com. 900 IN SRV 20 0 5269 alt2.xmpp-server.l.google.com.  
_xmpp-server._tcp.gmail.com. 900 IN SRV 20 0 5269 alt4.xmpp-server.l.google.com.  
_xmpp-server._tcp.gmail.com. 900 IN SRV 20 0 5269 alt1.xmpp-server.l.google.com.  
_xmpp-server._tcp.gmail.com. 900 IN SRV 5 0 5269 xmpp-server.l.google.com.  
_xmpp-server._tcp.gmail.com. 900 IN SRV 20 0 5269 alt3.xmpp-server.l.google.com.
```

- ¿Por qué se usan tan poco? ¿Por qué no se usan para la Web?

RR de tipo TXT

- Permite asociar texto con un nombre
- Una forma de añadir funcionalidad a DNS sin nuevos RRs
- Ejemplos de aplicación:
 - *Sender Policy Framework (SPF)*:
 - Validar qué máquinas de un dominio pueden enviar correo
 - NOTA: existe también un RR de tipo SPF
 - Verificar la propiedad de un dominio para Google
- Ejemplo de consulta: *dig fi.upm.es. TXT*

```
fi.upm.es.   86400 IN    TXT    "v=spf1 ip4:138.100.8.0/24 ip4:138.100.198.0/24 ip4:138.100.4.67 -all"  
fi.upm.es.   86400 IN    TXT    "google-site-verification=rJT2Yatvyg4HepVHZ-  
nk6LrxHNNIArZHaNxhkFCSgU"
```

RR de tipo NS (Name Server)

- Primer uso: especificar servidores de nombres para un dominio
- Ejemplo: *dig fi.upm.es. NS*

```
fi.upm.es.   86400 IN    NS    chita.fi.upm.es.  
fi.upm.es.   86400 IN    NS    zape.fi.upm.es.  
fi.upm.es.   86400 IN    NS    tarzan.fi.upm.es.  
fi.upm.es.   86400 IN    NS    galileo.ccupm.upm.es.  
fi.upm.es.   86400 IN    NS    ns.fi.upm.es.
```

- Ejemplo: *dig 243.100.138.in-addr.arpa. NS*

```
243.100.138.in-addr.arpa. 86400 IN    NS    zape.fi.upm.es.  
243.100.138.in-addr.arpa. 86400 IN    NS    chita.fi.upm.es.  
243.100.138.in-addr.arpa. 86400 IN    NS    galileo.ccupm.upm.es.  
243.100.138.in-addr.arpa. 86400 IN    NS    tarzan.fi.upm.es.  
243.100.138.in-addr.arpa. 86400 IN    NS    ns.fi.upm.es.
```

Servidores de nombres en UPM

- Ejemplo: *dig upm.es. NS*
upm.es. 7054 IN NS galileo.ccupm.upm.es.
upm.es. 7054 IN NS einstein.ccupm.upm.es.
upm.es. 7054 IN NS sun.rediris.es.
upm.es. 7054 IN NS chico.rediris.es.
- Ejemplo: *dig etsia.upm.es. NS*
etsia.upm.es. 42359 IN NS einstein.ccupm.upm.es.
etsia.upm.es. 42359 IN NS galileo.ccupm.upm.es.
- Ejemplo: *dig fi.upm.es. NS*
fi.upm.es. 86400 IN NS chita.fi.upm.es.
fi.upm.es. 86400 IN NS zape.fi.upm.es.
fi.upm.es. 86400 IN NS tarzan.fi.upm.es.
fi.upm.es. 86400 IN NS galileo.ccupm.upm.es.
fi.upm.es. 86400 IN NS ns.fi.upm.es.
- Ejemplo: *dig datsi.fi.upm.es NS*
datsi.fi.upm.es. 86400 IN NS ns.fi.upm.es.
datsi.fi.upm.es. 86400 IN NS chita.fi.upm.es.
datsi.fi.upm.es. 86400 IN NS galileo.ccupm.upm.es.
datsi.fi.upm.es. 86400 IN NS tarzan.fi.upm.es.
datsi.fi.upm.es. 86400 IN NS zape.fi.upm.es.

RR de tipo NS para delegación

- Segundo uso: delegar subdominio a s.nombres (pto. montaje)
 - Aparece como nombre del RR el del subdominio
 - Lista subdominios delegados no se puede obtener mediante consulta
- Ejemplo: UPM delega administración de sus recursos DNS a FI:
 - fichero de zona de *upm.es.* debe incluir:

```
fi.upm.es. 86400 IN NS chita.fi.upm.es.  
fi.upm.es. 86400 IN NS zape.fi.upm.es.  
fi.upm.es. 86400 IN NS tarzan.fi.upm.es.  
fi.upm.es. 86400 IN NS galileo.ccupm.upm.es.  
fi.upm.es. 86400 IN NS ns.fi.upm.es.
```

- fichero de zona de *100.138.in-addr.arpa.* debe incluir:

```
243.100.138.in-addr.arpa. 86400 IN NS zape.fi.upm.es.  
243.100.138.in-addr.arpa. 86400 IN NS chita.fi.upm.es.  
243.100.138.in-addr.arpa. 86400 IN NS galileo.ccupm.upm.es.  
243.100.138.in-addr.arpa. 86400 IN NS tarzan.fi.upm.es.  
243.100.138.in-addr.arpa. 86400 IN NS ns.fi.upm.es.
```

Glue records

- Posibles círculos viciosos en la traducción de nombres
- Si s. nombres de subdominio (Ssub) pertenece a subdominio
 - P.e. en dominio *upm.* SN de *fi.*: *chita.fi.upm.es.*, *ns.fi.upm.es.*, ...
 - Para obtener IP de cualquier máq. subdominio → contactar con Ssub
 - ¿IP de *www.fi.upm.es.*? → contactar con *chita.fi.upm.es.*
 - Pero para hacerlo necesito IP de Ssub → contactar con Ssub
 - ¿IP de *chita.fi.upm.es.*? → contactar *chita.fi.upm.es.*
- Glue record (GR)
 - RR de tipo A/AAAA que se incluye en un dominio ajeno
 - Solución c. vicioso: padre debe incluir RR tipo A con dir. IP de Ssub
 - Aumenta problemas de coherencia
 - Cambios en IP de Ssub deben reflejarse también en dominio padre
 - No necesario *glue record* para servidor externo o en dominio padre
 - Siempre se puede obtener su traducción

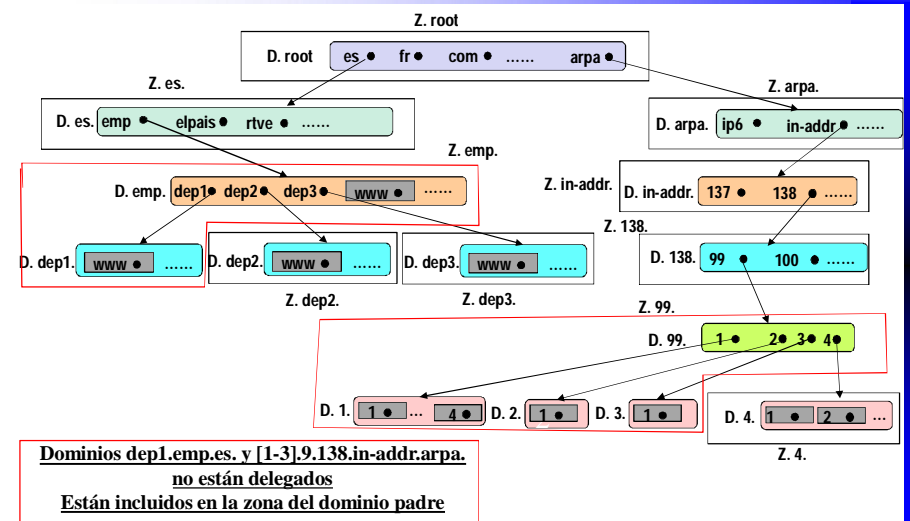
Delegaciones y Glue Records en UPM

- upm.es.*
etsia.upm.es. 42359 IN NS einstein.ccupm.upm.es.
etsia.upm.es. 42359 IN NS galileo.ccupm.upm.es.
fi.upm.es. 86400 IN NS chita.fi.upm.es.
fi.upm.es. 86400 IN NS zape.fi.upm.es.
fi.upm.es. 86400 IN NS tarzan.fi.upm.es.
fi.upm.es. 86400 IN NS galileo.ccupm.upm.es.
fi.upm.es. 86400 IN NS ns.fi.upm.es.
chita.fi.upm.es. 86400 IN A 138.100.8.23 ; Glue Record
tarzan.fi.upm.es. 86400 IN A 138.100.8.6 ; Glue Record
zape.fi.upm.es. 86400 IN A 138.100.8.1 ; Glue Record
ns.fi.upm.es. 86400 IN A 138.100.8.11 ; Glue Record
- fi.upm.es*
datsi.fi.upm.es. 86400 IN NS ns.fi.upm.es.
datsi.fi.upm.es. 86400 IN NS chita.fi.upm.es.
datsi.fi.upm.es. 86400 IN NS galileo.ccupm.upm.es.
datsi.fi.upm.es. 86400 IN NS tarzan.fi.upm.es.
datsi.fi.upm.es. 86400 IN NS zape.fi.upm.es.

Ejemplo hipotético

- Empresa con sede central y tres departamentos
 - Un administrador gestiona sede central, dep1 y dep2
 - Dep3 con administrador propio (y servidor de correo propio)
- Detalles de servidores de nombres de cada dominio:
 - Sede central de la empresa (emp.es.): 2 s. de nombres
 - maestro en dominio (ns.emp.es.); esclavo externo (ns.isp.com.)
 - Dep1 (dep1.emp.es.): subdominio no delegado
 - Dep2 (dep2.emp.es.): subdominio delegado a mismos servidores
 - Dep3 (dep3.emp.es.): 3 s. de nombres
 - Maestro (ns1.dep3.emp.es.); esclavo interno (ns2) y externo (ns.isp.com.)
- Empresa tiene asignada red clase B 138.99.0.0
 - 138.99.1 central; 138.99.2 dep1; 138.99.3 dep2; 138.99.4 dep3
 - Sólo está delegado subdominio de 138.99.4

Jerarquía de zonas ej. hipotético



F. zona emp.es. (ns.emp.es.)

emp.es.		IN	SOA	ns.emp.es.
emp.es.	86400	IN	NS	ns.emp.es. ; servidor maestro del dominio
emp.es.	86400	IN	NS	ns.isp.com. ; servidor esclavo externo
; dep2 delegado a mismos servidores				
dep2.emp.es.	86400	IN	NS	ns.emp.es. ; servidor maestro en el dominio padre
dep2.emp.es.	86400	IN	NS	ns.isp.com. ; servidor esclavo externo
; dep3 delegado a servidor maestro en el subdominio				
dep3.emp.es.	86400	IN	NS	ns1.dep3.emp.es. ; servidor maestro en su propio subdominio
dep3.emp.es.	86400	IN	NS	ns2.dep3.emp.es. ; servidor esclavo en su propio subdominio
dep3.emp.es.	86400	IN	NS	ns.isp.com. ; servidor esclavo externo
emp.es.	86400	IN	MX	10 mail1.emp.es. ; servidor de correo preferente para la empresa
emp.es.	86400	IN	MX	20 mail2.emp.es. ; servidor de correo de reserva para la empresa
dep1.emp.es.	86400	IN	MX	10 mail1.emp.es. ; servidor de correo preferente para dep1
dep1.emp.es.	86400	IN	MX	20 mail2.emp.es. ; servidor de correo de reserva para dep1
; Máquinas en el dominio de la empresa				
ns.emp.es.	86400	IN	A	138.99.1.1
mail1.emp.es.	86400	IN	A	138.99.1.2
mail2.emp.es.	86400	IN	A	138.99.1.3
www.emp.es.	86400	IN	A	138.99.1.4
www.dep1.emp.es.	86400	IN	A	138.99.2.1; RR de subdominio no delegado en misma zona
; Glue records para subdominio dep3				
ns1.dep3.emp.es.	86400	IN	A	138.99.4.1
ns2.dep3.emp.es.	86400	IN	A	138.99.4.2

F. zona dep2.emp.es. (ns.emp.es.)

dep2.emp.es.		IN	SOA	ns.emp.es.
dep2.emp.es.	86400	IN	NS	ns.emp.es. ; servidor maestro del dominio (=padre)
dep2.emp.es.	86400	IN	NS	ns.isp.com. ; servidor esclavo externo
; Correo				
dep2.emp.es.	86400	IN	MX	10 mail1.emp.es. ; s. correo preferente para dep2
dep2.emp.es.	86400	IN	MX	20 mail2.emp.es. ; s. correo de reserva para dep2
; Máquinas en el dominio de dep2				
www.dep2.emp.es.	86400	IN	A	138.99.3.1
backup.dep2.emp.es.	86400	IN	CNAME	www.dep2.emp.es.

F. zona dep3.emp.es. (ns1.dep3.emp.es.)

dep3.emp.es.		IN	SOA	ns1.dep3.emp.es
dep3.emp.es.	86400	IN	NS	ns1.dep3.emp.es.; <i>servidor maestro del dominio (!=padre)</i>
dep3.emp.es.	86400	IN	NS	ns2.dep3.emp.es.; <i>servidor esclavo en el propio dominio</i>
dep3.emp.es.	86400	IN	NS	ns.isp.com.; <i>servidor esclavo externo</i>
; <i>Correo</i>				
dep3.emp.es.	86400	IN	MX	10 mail.dep3.emp.es.; <i>s. correo preferente para dep3</i>
dep3.emp.es.	86400	IN	MX	20 mail2.emp.es.; <i>s. correo de reserva para dep3</i>
; <i>Máquinas en el dominio de dep3</i>				
ns1.dep3.emp.es.	86400	IN	A	138.99.4.1
ns2.dep3.emp.es.	86400	IN	A	138.99.4.2
mail.dep3.emp.es.	86400	IN	A	138.99.4.3
www.dep3.emp.es.	120	IN	A	138.99.4.4; <i>reparto de carga en servicio web</i>
www.dep3.emp.es.	120	IN	A	138.99.4.5; <i>reparto de carga en servicio web</i>

F. zona 99.138. (ns.emp.es.)

99.138.in-addr.arpa.		IN	SOA	ns.emp.es
99.138.in-addr.arpa.	86400	IN	NS	ns.emp.es.
99.138.in-addr.arpa.	86400	IN	NS	ns.isp.com.
; <i>dir. IP de dep3 delegadas a servidor maestro en el subdominio (no se necesitan glue records)</i>				
4.99.138.in-addr.arpa.	86400	IN	NS	ns1.dep3.emp.es.
4.99.138.in-addr.arpa.	86400	IN	NS	ns2.dep3.emp.es.
4.99.138.in-addr.arpa.	86400	IN	NS	ns.isp.com.
; <i>Máquinas de sede central, dep1 y dep2</i>				
1.1.99.138.in-addr.arpa.	86400	IN	PTR	ns.emp.es.
2.1.99.138.in-addr.arpa.	86400	IN	PTR	mail1.emp.es.
3.1.99.138.in-addr.arpa.	86400	IN	PTR	mail2.emp.es.
4.1.99.138.in-addr.arpa.	86400	IN	PTR	www.emp.es.
1.2.99.138.in-addr.arpa.	86400	IN	PTR	www.dep1.emp.es.
1.3.99.138.in-addr.arpa.	86400	IN	PTR	www.dep2.emp.es.

F. zona 4.99.138. (ns1.dep3.emp.es.)

4.99.138.in-addr.arpa.		IN	SOA	ns1.dep3.emp.es.
4.99.138.in-addr.arpa.	86400	IN	NS	ns1.dep3.emp.es.
4.99.138.in-addr.arpa.	86400	IN	NS	ns2.dep3.emp.es.
4.99.138.in-addr.arpa.	86400	IN	NS	ns.isp.com.
; <i>Máquinas de dep3</i>				
1.4.99.138.in-addr.arpa.	86400	IN	PTR	ns1.dep3.emp.es.
2.4.99.138.in-addr.arpa.	86400	IN	PTR	ns2.dep3.emp.es.
3.4.99.138.in-addr.arpa.	86400	IN	PTR	mail.dep3.emp.es.
4.4.99.138.in-addr.arpa.	86400	IN	PTR	www.dep3.emp.es.
5.4.99.138.in-addr.arpa.	86400	IN	PTR	www.dep3.emp.es.

Servidores de nombres raíces

- Hay "13" servidores de dominio raíz (.) replicados
 - Desde *a.root-servers.net* hasta *m.root-servers.net*
 - "13" porque esa información cabe en paquete UDP
 - DNS usa UDP (53); y sólo TCP(53) cuando tamaño lo aconseja
 - ¿Problemas de escalabilidad?
 - Detrás de cada uno hay múltiples servidores (uso de *anycast*)
 - Incluyen NS y *glue records* de dominios de nivel 1º (TLDs)
 - Aunque también gestionan algunos dominios primer nivel → zona *arpa*.
 - Cada serv. DNS tiene dir. de servidores raíz (fichero *root.servers*)
 - Se debe actualizar periódicamente
- Lista y localización: <http://root-servers.org>

Servidores DNS

- Servidor gestiona (*authoritative*) $N (\geq 0)$ zonas directas/inversas
 - De algunas puede ser maestro de otras esclavo
 - Si $N=0 \rightarrow$ servidor sólo caché
- Servidor DNS puede tener doble rol (algo confuso):
 - Siempre proporciona acceso a sus zonas
 - excepto si $N=0$, que no tiene zonas
 - Opcional puede participar en navegación recursiva
 - Si recibe petición que no pertenece a su zona, contacta con otro servidor
 - En vez de responder a solicitante con información de por dónde seguir
 - Usará una caché para ir guardando resultados de peticiones
- Servidor debe ofrecer navegación iterativa; recursiva opcional
 - **Cliente siempre debe interactuar con servidor recursivo**
 - Si sólo caché, debería ofrecer recursiva (si no, sin sentido)
- Servidor puede limitar qué clientes le pueden enviar peticiones
 - Por seguridad sólo peticiones de ciertas máquinas

Diversos tipos de servidores DNS

- Serv. *authoritative* no recursivo
 - Da sólo acceso iterativo a sus zonas para todo el mundo
 - p.e. servidor raíz o de TLD
- Servidor *authoritative* y recursivo sólo para clientes internos
 - Da acceso iterativo a sus zonas para todo el mundo
 - Sirve todas la peticiones pero sólo de los clientes de la organización
 - Con caché
 - P.e. servidores de la escuela: *chita.fi.upm.es*,...
- S. sólo caché (*nonauthoritative*) recursivo para clientes internos
 - Sirve a clientes de una organización actuando como caché
 - *DNS-proxy*, *DNS-forwarder*,...
- Servidor sólo caché (*nonauthoritative*) recursivo público
 - Sirve a cualquier cliente
 - <https://www.lifewire.com/free-and-public-dns-servers-2626062>

Resolver

- Parte cliente de DNS: da servicio a aplicaciones en un nodo
- Implementado habitualmente como biblioteca ($C \subset$ en *libc*)
- Proporciona API para traducción directa e inversa:
 - UNIX: *gethostbyname/getaddrinfo* y *gethostbyaddr/getnameinfo*
- Configurado con servidores de nombres a los que consulta
 - Requiere también las direcciones IP de todos esos s. de nombres
 - En UNIX: */etc/resolv.conf*
- Esos servidores de nombres deben ser recursivos
 - *Resolver* no sabe navegar
- Puede usar caché (las aplicaciones también)
- UNIX permite configurar mecanismo de traducción de *hosts*
 - */etc/hosts*, DNS, NIS, LDAP (*/etc/nsswitch.conf*)

Resolución de consultas

- Servidor *S* recibe una consulta *C* de *N*:
 - Compara con RRs de todas sus zonas y de su caché (si usa)
 - Selecciona mejor encaje \rightarrow RR (*RRX*) que sea sufijo más largo de *C*
 - Si encaje completo \rightarrow envía a *N* consuelta resuelta
 - La marca como *authoritative* si no proviene de la caché
 - Si varios RRs satisfacen consulta, se envía a *N* lista con todos
 - Servidor rota la lista cada vez (*Round-robin DNS*) para reparto de carga
 - Se incluye información adicional para agilizar la operación
 - P.e. consulta MX puede retornar los RRs de tipo A de servidores de correo
 - Si encaje no completo, *RRX* \rightarrow NSs de dominio por donde continuar
 - En el peor caso, los NSs de servidores raíz
 - Si op. recursiva: *S* envía consulta a uno de los NSs encontrados
 - Si op. no recursiva: *S* envía a *N* los RRs de los NSs encontrados
 - Si *S* conoce direcciones de NSs encontrados
 - Si no recursiva: las incluye como info. adicional en mens. de respuesta a *N*
 - Si recursiva: *S* las usa para contactar; sino tiene que obtenerlas

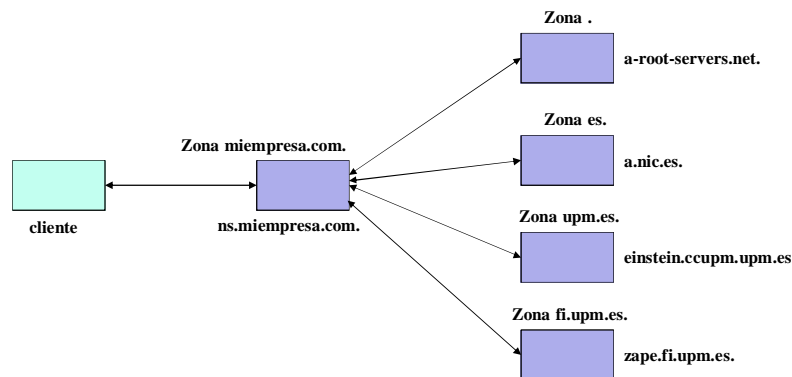
Ejemplos reales de traducción

- Operación de traducción directa: *www.fi.upm.es*.
- Operación de traducción inversa: *138.100.243.10*.
- Resolver tiene configurado como SN (3 opciones):
 - SN1: *ns.miempresa.com.*; IP 139.100.1.1
 - Zonas gestionadas: *miempresa.com.* y *100.139.in-addr.arpa.*, ...
 - SN2: *einstein.ccupm.upm.es.*; IP 138.100.4.8
 - Zonas gestionadas: *upm.es.* y *100.138.in-addr.arpa.*, ...
 - SN3: *zape.fi.upm.es.*; IP 138.100.8.1
 - Zonas gestionadas: *fi.upm.es.* y *243.100.138.in-addr.arpa.*, ...
- Supuestos:
 - traducción recursiva resolver-SN e iterativa desde SN
 - cachés vacías
- Test online de traducciones:
 - <https://dnsquery.org/>

Traducción directa usando SN1

- Aplicación llama a *gethostbyname("www.fi.upm.es.")* de resolver
- Resolver envía petición DNS de tipo A a dir. de SN1: 139.100.1.1
- SN1 mejor encaje: *.* → elige un s. raíz: *a.root-servers.net.* (198.41.0.4)
 - a.root-servers.net.* mejor encaje: *es.* → envía a SN1 los NSs de *es.*
 - Y sus *glue records* como información adicional
- SN1 elige *a.nic.es.* (194.69.254.1)
 - a.nic.es.* mejor encaje: *upm.es.* → envía a SN1 los NSs de *upm.es.*
 - Y sus *glue records* como información adicional
- SN1 elige *einstein.ccupm.upm.es.* (138.100.4.8)
 - einstein.ccupm.upm.es.* mejor encaje: *fi.upm.es.*
 - envía a SN1 los NSs de *fi.upm.es.* y sus *glue records*
- SN1 elige *zape.fi.upm.es.* (138.100.8.1)
 - zape.fi.upm.es.* Encaje completo: *www.fi.upm.es.*
 - envía a SN1 el NS de tipo A → *www.fi.upm.es.* | 138.100.243.10
 - SN1 se lo envía al resolver y éste retorna la IP a la aplicación

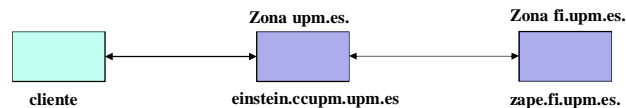
Traducción directa con SN1



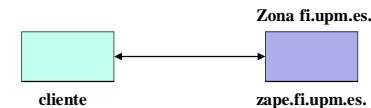
Traducción directa usando SN2 y SN3

- Aplicación llama a *gethostbyname("www.fi.upm.es.")* de resolver
- Resolver envía petición DNS de tipo A a dir. de SN2: 139.100.4.8
- SN2 mejor encaje: *fi.upm.es.* → elige *zape.fi.upm.es.* (138.100.8.1)
 - zape.fi.upm.es.* encaje completo: *www.fi.upm.es.*
 - envía a SN2 el NS de tipo A → *www.fi.upm.es.* (138.100.243.10)
 - SN2 se lo envía al resolver y éste retorna la IP a la aplicación
- Aplicación llama a *gethostbyname("www.fi.upm.es.")* de resolver:
- Resolver envía petición DNS de tipo A a dir. de SN3: 138.100.8.1
- SN3 encaje completo: *www.fi.upm.es.* (138.100.243.10)
 - SN3 se lo envía al resolver y éste retorna la IP a la aplicación
- ¿Y si hubiera elegido *galileo* en vez de *einstein*?
 - Un paso menos en la traducción ya que es secundario de zona *fi*
- ¿Y si hay que traducir *www.datsi.fi.upm.es.*?
 - Mismos pasos ya que *zape* gestiona también zona *datsi*

Traducción directa con SN2



Traducción directa con SN3



Mantenimiento de réplicas

- Sincronización de secundario con primario
 - Sigue esquema *pull*: esclavo pide información de zona a maestro
 - Periódicamente (tal como lo especifica SOA)
 - O cuando maestro avisa de cambios (*NOTIFY*; extensión opcional)
 - Si cambio: transferencia zona completa (*AXFR*) o incremental (*IXFR*)
 - *IXFR* (extensión opcional): sólo cambios entre n° serie maestro y esclavo
 - Sólo se debe permitir transferencia de zona entre maestro y esclavos
- PAEL: no asegura consistencia en ningún escenario
 - Red partida, secundario aislado sigue sirviendo peticiones
 - Valores obsoletos durante un tiempo limitado (tal como especifica SOA)
 - Red normal, actualización primario pero secundario sirve peticiones
 - Valores obsoletos hasta sincronización de secundario con primario
 - Además, cachés de clientes y de SNs sin consistencia
 - Valores obsoletos durante tiempo limitado (tal como especifica TTL)

Actualización de DNS

- Por defecto (y en RFC original), administración manual local
 - Editar fichero zona
 - Incrementando n° en SOA
 - Avisando a proceso maestro para que relea fichero de zona
 - p.e. enviándole una señal
- *Dynamic DNS*
 - Protocolo DNS incluye ops. para actualizar zona
 - Añadir, modificar y borrar RR pero no crear nuevas zonas
 - Mucho más flexible pero menos seguro
 - Mandato *nsupdate*
 - Algunas aplicaciones:
 - Permitir que máquinas mantengan mismo nombre en sistemas DHCP
 - Servidor elige cualquier puerto y usa SRV (requerido *Active Directory*)

Lightweight Directory Access Protocol

- Precedente: X.500 servicio de directorio de ISO
 - Concebido para ser un directorio mundial
 - Complejo
 - Pesado: Ejecuta sobre la pila OSI
 - Protocolo de acceso DAP (*Directory Access Protocol*)
- LDAP (*Lightweight Directory Access Protocol*, RFC 4510)
 - Basado en X.500
 - Más sencillo
 - Más ligero: ejecuta sobre la pila TCP/IP
 - Es solo un protocolo pero define implícitamente un modelo de datos
 - No define aspectos de implementación
 - Distintos sistemas ofrecen una interfaz LDAP (p.e. *Active Directory*)
 - Actualmente versión 3

Objetos y clases

- Entidad → Objeto (entrada) en LDAP
 - Orientado a objetos: Objeto ∈ Clase (atributo *objectClass*)
- Clase define conjunto de atributos del objeto
 - Tipo del atributo | obligatorio(ob) u optativo(op) | valor único o múltiple
- Herencia: clases forman una jerarquía (*top* raíz de jerarquía)
 - Clase derivada hereda atributos de superclases
- Tipos de clases:
 - Abstracta (AB): no pueden definirse objetos de esa clase (p.e. *top*)
 - Estructural (ES): Objeto ∈ Una y solo una clase estructural
 - No puede cambiar la clase estructural de un objeto
 - Auxiliar (AU): Objeto puede estar asociado a varias clases auxiliares
 - Pueden añadirse dinámicamente: Facilitan extensión de objetos
 - Superclase(ES)=ES|AB; Superclase(AU)=AU|AB

Ejemplos de clases

- **top**: raíz; **AB**; **ob**: *objectClass*
- **person**: ↓*top*; **ES**; **ob**: *cn*, *sn*; **op**: *telephoneNumber*, ...
- **residentialPerson**: ↓*person*; **ES**; **ob**: *l*; **op**: *postalAddress*, ...
- **organization**: ↓*top*; **ES**; **ob**: *o*; **op**: *postalAddress*, ...
- **organizationalUnit**: ↓*top*; **ES**; **ob**: *ou*; **op**: *postalAddress*, ...
- **dcObject**: ↓*top*; **AU**; **ob**: *dc* (valor único)
- **device**: ↓*top*; **ES**; **ob**: *cn*; **op**: *serialNumber*, *o*, *ou*, *owner*, ...
- **groupOfNames**: ↓*top*; **ES**; **ob**: *cn*, *member*; **op**: *o*, *ou*, ...
- **alias**: ↓*top*; **ES**; **ob**: *aliasedObjectName*
- **referral**: ↓*top*; **ES**; **ob**: *ref*

Extracto de mi entrada en LDAP de FI

Formato de texto LDIF (LDAP Data Interchange Format): protocolo LDAP es binario

objectClass: inetOrgPerson	← estructural (<i>top</i> → <i>person</i> → <i>organizationalPerson</i> → <i>inetOrgPerson</i>)
objectClass: posixAccount	← auxiliar (<i>top</i> → <i>posixAccount</i>)
objectClass: fiEmployee	← auxiliar (<i>top</i> → <i>irisPerson</i> → <i>fiPerson</i> → <i>fiEmployee</i>)
objectClass: sambaSamAccount	← auxiliar (<i>top</i> → <i>sambaSamAccount</i>)

cn: Fernando Perez Costoya	}	}	}	<i>inetOrgPerson</i>
cn: F. P. Costoya				
sn: Perez Costoya				
telephoneNumber: 913367377				
mail: fperez@fi.upm.es				
uid: fperez				

uidnumber:	}	<i>posixAccount</i>
gidNumber:		

irisUserStatus: Activo	}	<i>irisPerson</i>	}	<i>fiEmployee</i>
fiRelationship: pdi				
fiTeaching:				

sambaSID:	}	<i>sambaSamAccount</i>

Extracto de entrada FI en LDAP de FI

objectClass: dcObject ← auxiliar (*top* → *dcObject*)
objectClass: organization ← estructural (*top* → *organization*)
objectClass: labeledURIObject ← auxiliar (*top* → *labeledURIObject*)
dc: fi ← atributo específico de *dcObject*
o:: RmFjdWx0YWQgZGUgSW5mb3Jtw6F0aWNhIC0gVVBVN
postalCode: 28660
l: Boadilla del Monte
st: Madrid
labeledURI: http://www.fi.upm.es ← atributo específico de *labeledURIObject*
telephoneNumber: +34 913367399

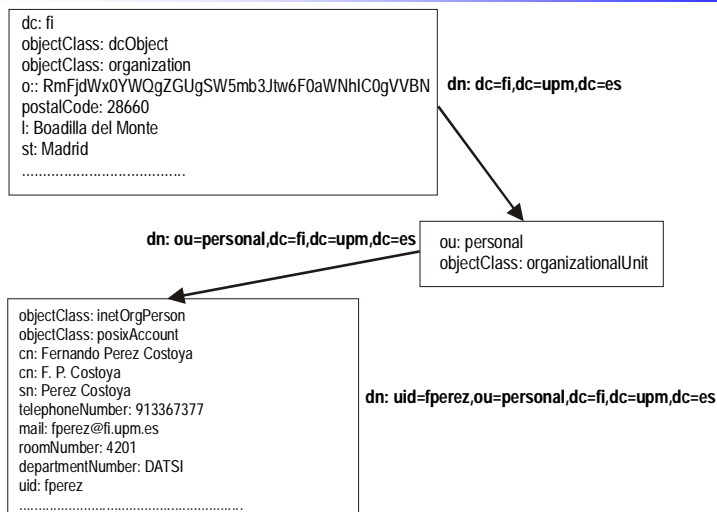
Decodificación de base 64

o: Facultad de Informática – UPM

Modelo de nombres

- Entrada tiene un nombre: *Relative Distinguished Name* (RDN)
 - 1 o más atributos de la entrada que la hacen única entre “hermanos”
 - *uid=fperez* (ej. múltiples: *cn=Fernando Perez Costoya+dni=76543210*)
- Jerarquía de nombres (*Directory Information Tree*, DIT)
 - Nombre completo (*path*): *Distinguished Name* (DN)
 - RDN de la entrada + DN del padre (separados por comas)
 - **dn**: *uid=fperez,ou=personal,dc=fi,dc=upm,dc=es*
 - No confundir con jerarquía de clases
 - Similar a SF pero directorios también tienen información asociada
 - Nombre del objeto raíz (sufijo o base): a discreción
 - Convenio: a partir de dominio DNS usando clase auxiliar *dcObject*
 - Dominio: *fi.upm.es* → **dn**: *dc=fi,dc=upm,dc=es*
 - Servidor LDAP gestiona 1 ó más DIT
 - Servidor devuelve metainformación en objetos/atrib. operacionales
 - DIT gestionados por el servidor, esquemas soportados, ...

Extracto de rama del DIT del LDAP de FI



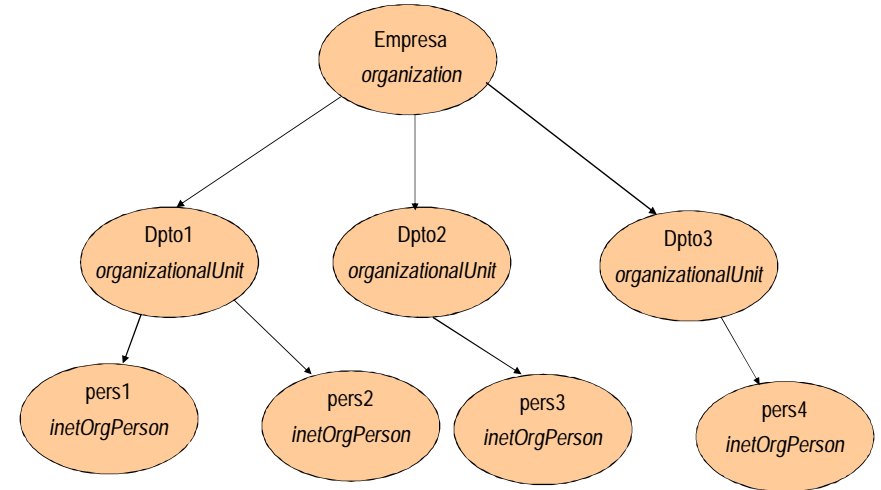
Distribución y replicación

- Espacio de nombres distribuido usando *referrals*
 - Objeto en DIT especifica punto de montaje
 - No definido el modelo de navegación
 - Implementación más habitual iterativa
 - Aunque también recursiva (*chaining*)
- Replicación de espacio de nombres no definida por estándar
 - OpenLDAP no garantiza coherencia (como DNS)
 - OpenLDAP admite dos esquemas:
 - Maestro-esclavo (asimétrico): 1 primario y N secundarios
 - Actualización en primario pero consulta a cualquier réplica
 - Propagación a réplicas (Modo *push* o *pull*)
 - Multi-maestro (simétrico): consulta y actualización a cualquier réplica
 - Las réplicas se actualizan de forma independiente
 - Necesidad de consolidar cambios potencialmente conflictivos

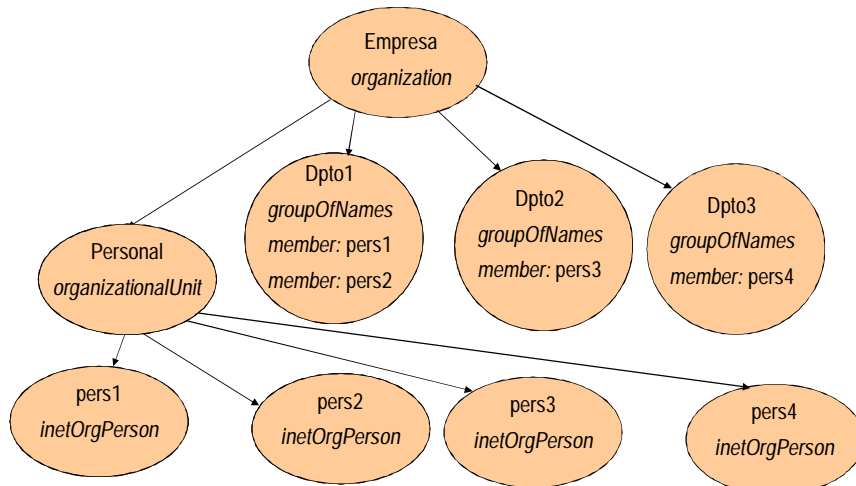
Diseño del DIT

- No trivial: requiere experiencia
- Análisis previo de info. del SD y cómo evolucionará
 - Diseño debería evitar que cambios previstos en info. modifiquen DIT
 - Cambio debería afectar a atributos en vez de a estructura de DIT
 - Mejor árbol poco profundo
- Ej.: empresa donde personal cambia de dpto. con frecuencia
 - Diseño 1
 - 1 *organizationalUnit*/dpto. + 1 *inetOrgPerson*/persona
 - Entrada de persona hija de entrada de su departamento
 - Diseño 2
 - 1 *organizationalUnit* para todo el personal + 1 *inetOrgPerson*/persona
 - 1 *groupOfNames*/dpto. con 1 atributo *member*/persona
 - Persona cambia de departamento: cambio atributos, no cambio DIT
 - Aunque ciertas búsquedas pueden ralentizarse

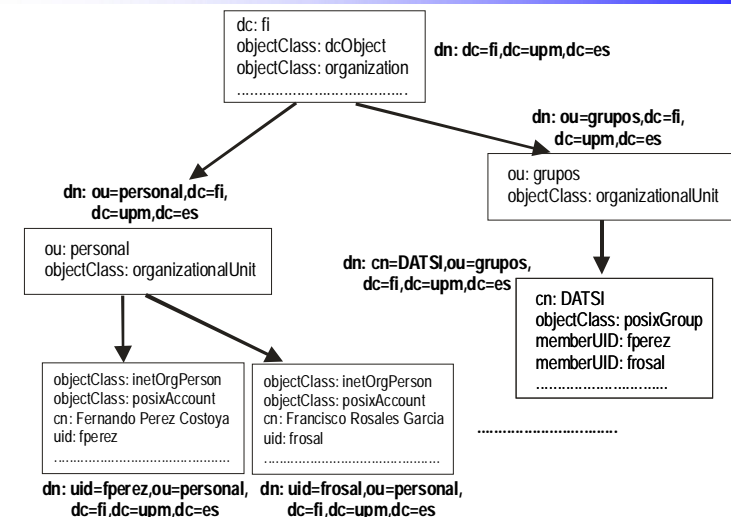
Diseño 1



Diseño 2



Extracto de jerarquía de LDAP de FI



Operaciones de LDAP

- *Bind/Unbind*: conecta y autentica/desconecta
- *Search*: realiza una búsqueda basada en los parámetros:
 - DN base de la búsqueda
 - Ámbito: Sólo la entrada base, sólo hijos o todo el sub-árbol
 - Filtro de búsqueda
 - Atributos que se devuelven (además, si valores o sólo tipos)
 - Si se siguen los alias o no durante la búsqueda
 - Limite de tiempo y máximo n° de entradas retornadas
- *Compare*: comprueba si DN dado tiene un valor en atributo
- *Add/Delete*: Añade/Elimina la entrada del DN dado
- *Modify*: Modifica atributos (añade, elimina o cambia) de un DN
- *Modify DN*: Cambia DN de una entrada
 - Renombra si sólo cambia RDN final; mueve en DIT en caso contrario

Acceso a operaciones de LDAP

- API de programación en C
 - *ldap_bind()*, *ldap_search()*, *ldap_add()*, *ldap_delete()*, *ldap_modify()*, ...
- Mandatos
 - *ldapsearch*, *ldapadd*, *ldapdelete*, *ldapmodify*, *ldapmodrdn*, ...
 - La mayoría usan el formato LDIF como entrada o salida
- Formato URL estándar para LDAP
 - *ldap://máquina:puerto/DNbase?atributos?ámbito?filtro*
 - *ldaps* si usa comunicación segura

Ejemplo de operaciones sobre diseño 1

- Alta de persona en la organización en el *dptoX*
 - *Add* nodo en *dptoX*
- Baja en organización de persona en el *dptoX*
 - *Delete* de nodo de *dptoX*
- Cambio de persona del *dptoX* al *dptoY*
 - *ModifyDN* de nodo de *dptoX* al *dptoY*
- Obtener n° teléfono de persona con un dirección de correo X
 - *Search*. **Base**: DN de la empresa; **Scope**: *default (subtree)*; **Filtro**: *(mail=X)*; **Atributos a recuperar**: *telephoneNumber*
- Obtener dirección de correo de todos los miembros de *dptoX*
 - *Search*. **Base**: DN de *dptoX*; **Scope**: *one*; **Filtro**: *NO*; **Atributos**: *mail*

Ejemplo de operaciones sobre diseño 2

- Alta de persona en la organización en el *dptoX*
 - *Add* en *personal* + *Modify* que haga persona *member* de *dptoX*
- Baja en organización de persona en el *dptoX*
 - *Modify* elimina persona como *member* de *dptoX* + *Delete* de *personal*
- Cambio de persona del *dptoX* al *dptoY*
 - 2 *Modify*: elimina como *member* de *dptoX* y lo añade como de *dptoY*
- Obtener n° teléfono de persona con un dirección de correo X
 - *Search*. **Base**: DN *personal*; **Scope**: *one*; **Filtro**: *(mail=X)*; **Atrib**: *tño*
- Obtener dirección de correo de todos los miembros de *dptoX*
 - Opción 1: 1 operación para obtener grupo y 1 por cada miembro:
 - *Search* (1). **Base**: DN *dptoX*; **Scope**: *base*; **Filtro**: *NO*; **Atrib**: *member*
 - *Search* (N). **Base**: DN *miembro*; **Scope**: *base*; **Filtro**: *NO*; **Atribut.**: *mail*
 - Opción 2, suponiendo que nodo persona incluye atributo con *dpto*.
 - *Search*. **Base**: DN *personal*; **Scope**: *one*; **Filtro**: *(dpto=X)*; **Atrib**: *mail*