

# Résolution de noms

cb

(/home/kouna/d01/adp/bcousin/Fute/Cours/Internet-2/15-DNS.fm- 9 Octobre 1998 12:05)

## PLAN

- Introduction
- Noms et adresses
- Principe de la résolution de noms
- La résolution de noms sous Internet/Unix
- Conclusion

## Bibliographie

- A.Fenyo, F.LeGuern, S.Tardieu, Se raccorder à Internet, Eyrolles, 1997
- L.Toutain, Internet et les réseaux locaux, Hermès, 1996.
- G.Hunt, TCP-IP network administration, O-Reilly, 1992.

## 1. Introduction


On a besoin d'un service mondial d'annuaire pour Internet.

Service associant le nom d'une station à son adresse IP :

- Par exemple :

 pondichery.irisa.fr  $\Rightarrow$  131.254.61.13

Mais aussi, il faut un service associant un serveur de messagerie au nom d'un utilisateur.

 bcousin@ifsic.univ-rennes1.fr  $\Rightarrow$  mercure.univ-rennes-1.fr

Il faut qu'il soit stable, fiable et performant.

## 2. Noms et adresses

### 2.1. Introduction

Les adresses IP sont adaptées à leurs tâches :

- identification dense (numérotation simple, sur 32 bits),
- aide à l'acheminement (netid + hostid)

Les êtres humains ont quelques difficultés à les utiliser :

- erreurs typographiques
- mémorisation difficile, etc.

La notation conventionnelle (décimale pointée) est insuffisante.

 On a besoin de noms symboliques, signifiants : `ma_machine`

. facile à administrer : `ma-machine.mon-entreprise.mon-pays`

Remarque : Problème similaire entre nom de fichier et référence interne au noyau du système de gestion des fichiers (“inode”).

Remarque : La distinction entre nom et adresse est artificielle, ils identifient tous les deux de manière unique un objet. L'un est dit logique (ou de haut niveau), l'autre physique (de bas niveau).

## 2.2. Espace plat d'adresses

Initialement sous Internet l'espace des noms était plat.

Un organisme central administre l'espace des noms :

- vérifie l'unicité du nom
- mémorise l'association nom/adresse
- NIC : "Internet Network Information Center"
- Avantages :
  - . les noms peuvent être quelconques, aussi courts que voulu
  - . la résolution d'adresse est triviale : consulter le fichier du site central
- Inconvénients
  - . conflits de noms fréquents
  - . le site central surchargé par les demandes de créations de noms et de modifications d'associations

La résolution de noms était peu performante :

- soit la résolution est centralisée, le site central est surchargé.
- soit la résolution est répartie, la distribution de la liste est extrêmement coûteuse.

## 2.3. Espace hiérarchique d'adresses

On résout les problèmes précédents au moyen d'un espace hiérarchique de noms et de la délégation.

### 2.3.1 L'espace des noms

- . Présente une structure **arborescente** (similaire au système de fichiers, par ex.)
- . Chaque noeud est identifié par un "label" :
  - d'au plus 63 caractères,
  - majuscules et minuscules étant identiques
- . Un noeud spécial, la racine, est identifié par une chaîne de caractères vides.
- . Le nom d'un noeud de l'arbre est identifié par la suite de labels rencontrés en partant de ce noeud et en allant vers la racine, chaque label étant séparé par un point.
  - un nom est unique, mais
  - plusieurs noms peuvent partager le même label
  - et un même nom peut avoir plusieurs labels identiques

Par exemple :

- ma-station.ifsic.univ-rennes1.fr

Dénomination relative ou absolue :

- . Un nom absolu possède (est terminé par) un point [convention dépendant de l'implémentation]  
par ex : ma-station.ifsic.univ-rennes1.fr.
- . Un nom relatif doit être complété localement par un suffixe, celui du domaine local  
par ex : ma-station(.ifsic.univ-rennes1.fr.)

Remarque : ne pas confondre sous-domaine (de noms) et sous-réseau (IP).

### 2.3.2 Délégation / Domaine de noms

Chaque organisation a la responsabilité de l'administration de son espace de noms (appelé “domain”).

Le responsable d'un domaine de noms peut décider de déléguer à un sous-responsable l'administration d'une partie du domaine des noms dont il a la charge.

Une arborescence de domaines et sous-domaines est ainsi créée.

Les sous-domaines d'un domaine de noms forment une partition du domaine.

Le nom d'un domaine de noms est formé par la suite des labels identifiant la branche associée au domaine de noms.

Remarque : The NIC (“Network Information Center”) a la responsabilité d'administrer le domaine de niveau le plus élevé.

Remarque : chaque domaine gère de manière autonome ses noms.

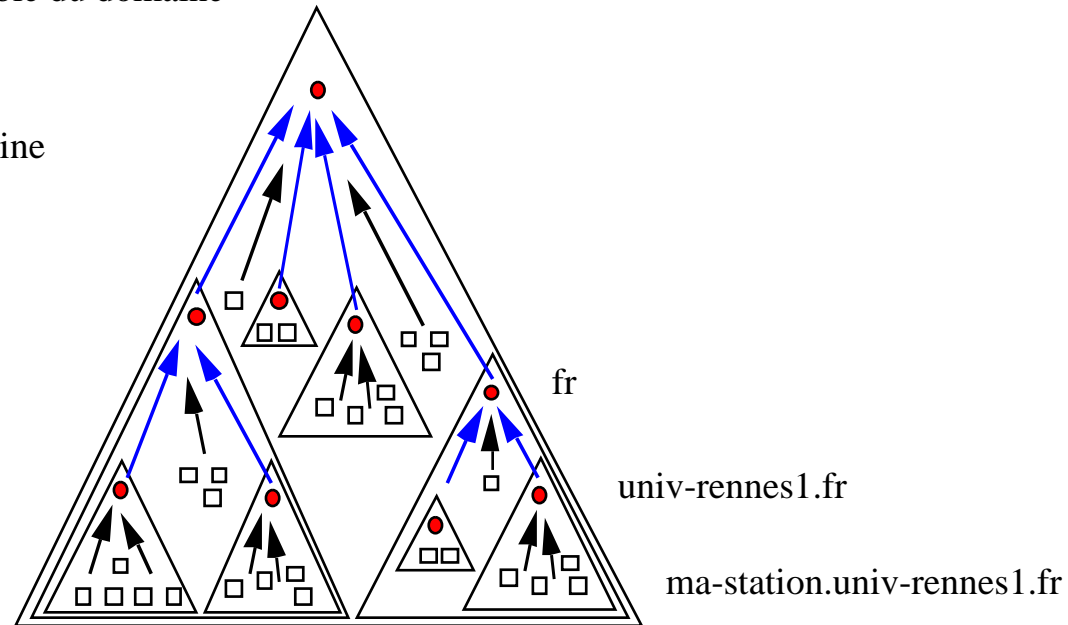
Remarque : l'arbre des domaines de noms n'a aucun besoin d'être équilibré.

Exemple :

● responsable du domaine

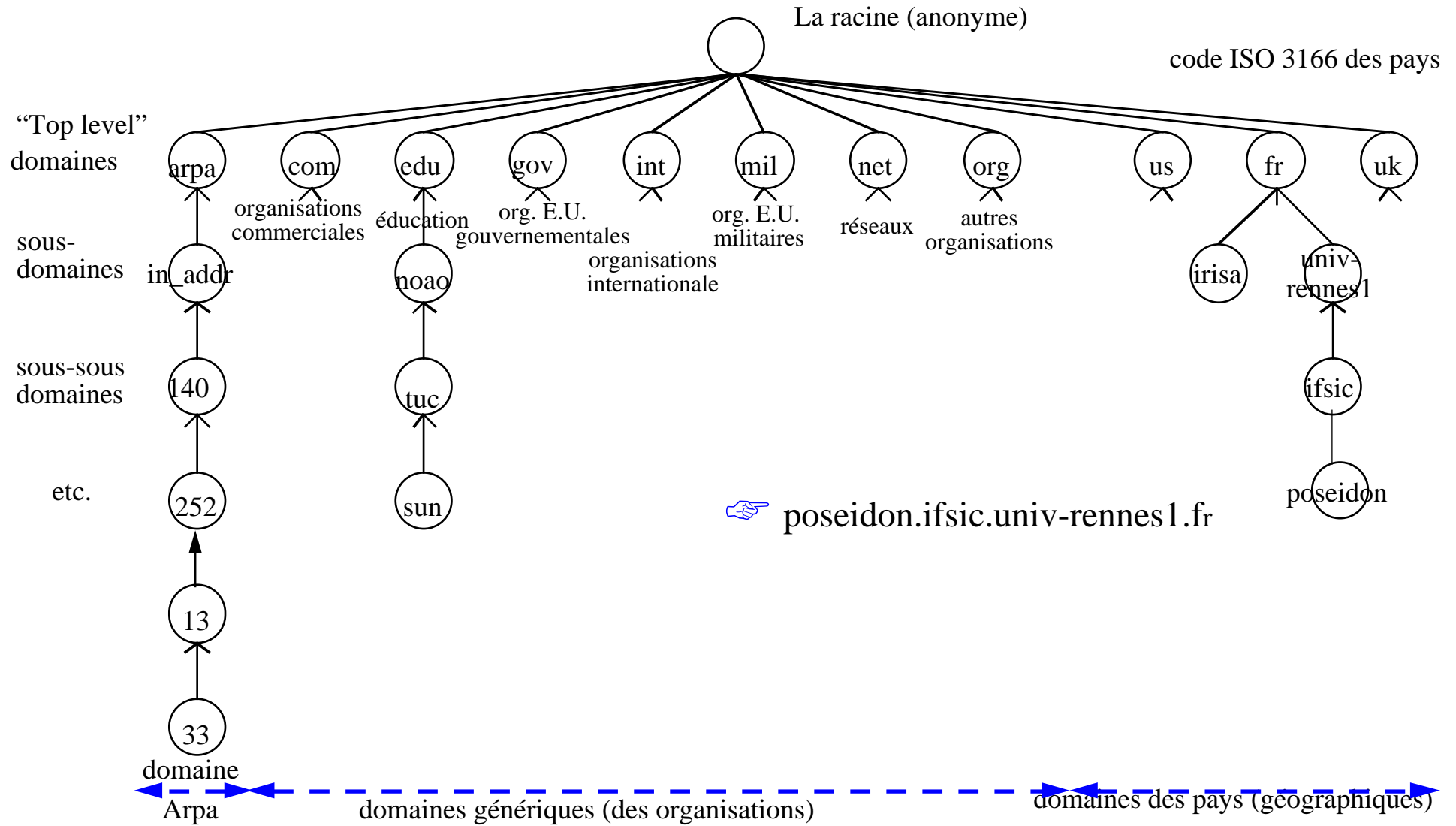
□ station

△ domaine





## 2.4. L'arborescence des domaines de noms sous Internet



### 3. La résolution de noms

#### 3.1. La “Host table”

##### 3.1.1 Présentation

Solution triviale au problème de traduction :


- chaque système contient un **fichier** des associations nom/adresse

Sous Unix : [/etc/hosts](#)

- Exemple : le fichier `/etc/hosts` de la station “poseidon” :

```
#
# Internet host table
#
127.0.0.1 localhost
148.60.4.20 poseidon.ifsic.univ-rennes1.fr poseidon
```

- La première entrée :
  - . 127.0.0.1 = “local loopback address”
  - . “localhost” = nom générique de la station elle-même

 les accès locaux ou distants peuvent être codés de manière identique
- La seconde entrée :
  - . l’adresse de la station : 148.60.4.20
  - . le nom et les **alias** de la station

### 3.1.2 Conclusion

Cette technique n'est pas "scalable" :

- le réseau mondial contient des millions de stations
  - . le fichier local devrait contenir une entrée pour toutes les stations
- le réseau mondial est modifié en permanence (pannes, adjonctions, suppressions, etc.)
  - . l'administration serait très difficile : mise-à-jour multiples, lenteur, risque d'incohérence, etc.

Il faut une technique offrant :

- un contrôle centralisé
- une dissémination automatique des associations : nom/adresse

☞ Système réparti de serveurs de noms : DNS

Cependant, cette technique locale est utilisée :

- lors du démarrage du système (lors du "boot"),
- sur les petits sites isolés du reste du réseau,
- sur les vieilles stations ne disposant de DNS.

## 3.2. Résolution des noms par serveurs

### 3.2.1 Les serveurs

#### “Domain Name Server”

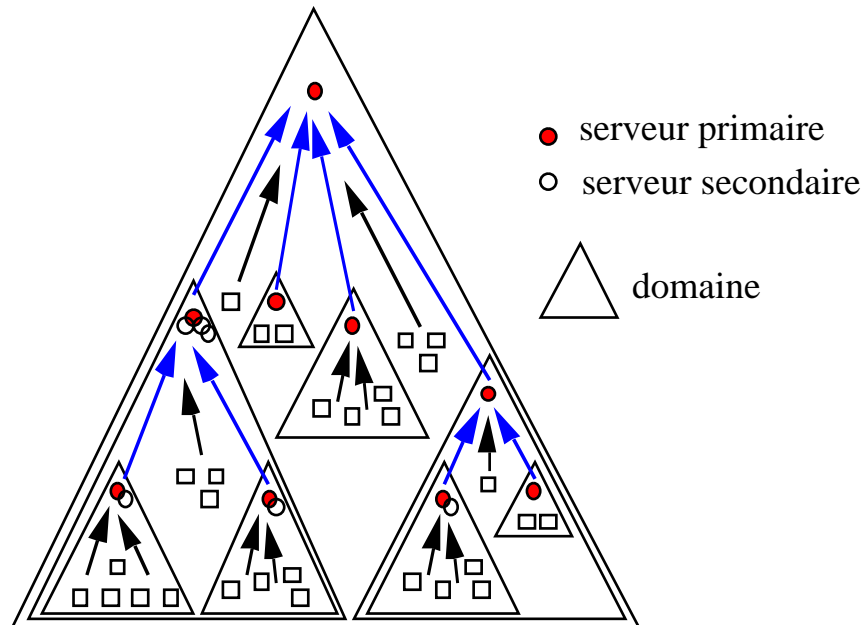
A chaque domaine de noms est associé au moins un serveur **primaire**.

- C’est le responsable du domaine (“authoritative server”)
  - . Les administrateurs ont un point central de gestion
- Ce serveur contient les informations relatives au domaine
  - . son père : le serveur du domaine supérieur (“root server”)
  - . ses fils : les serveurs de chacun de ses sous-domaines
  - . les noms qu’il gère directement

Ce serveur primaire peut être flanqué de serveurs **secondaires** (aucun ou plusieurs)

- ils assurent la permanence du service : redondance
- ils échangent automatiquement une copie des infos détenues par le primaire
  - . les informations obtenues à partir de ses serveurs sont à jour (“authoritative”).

Exemple :

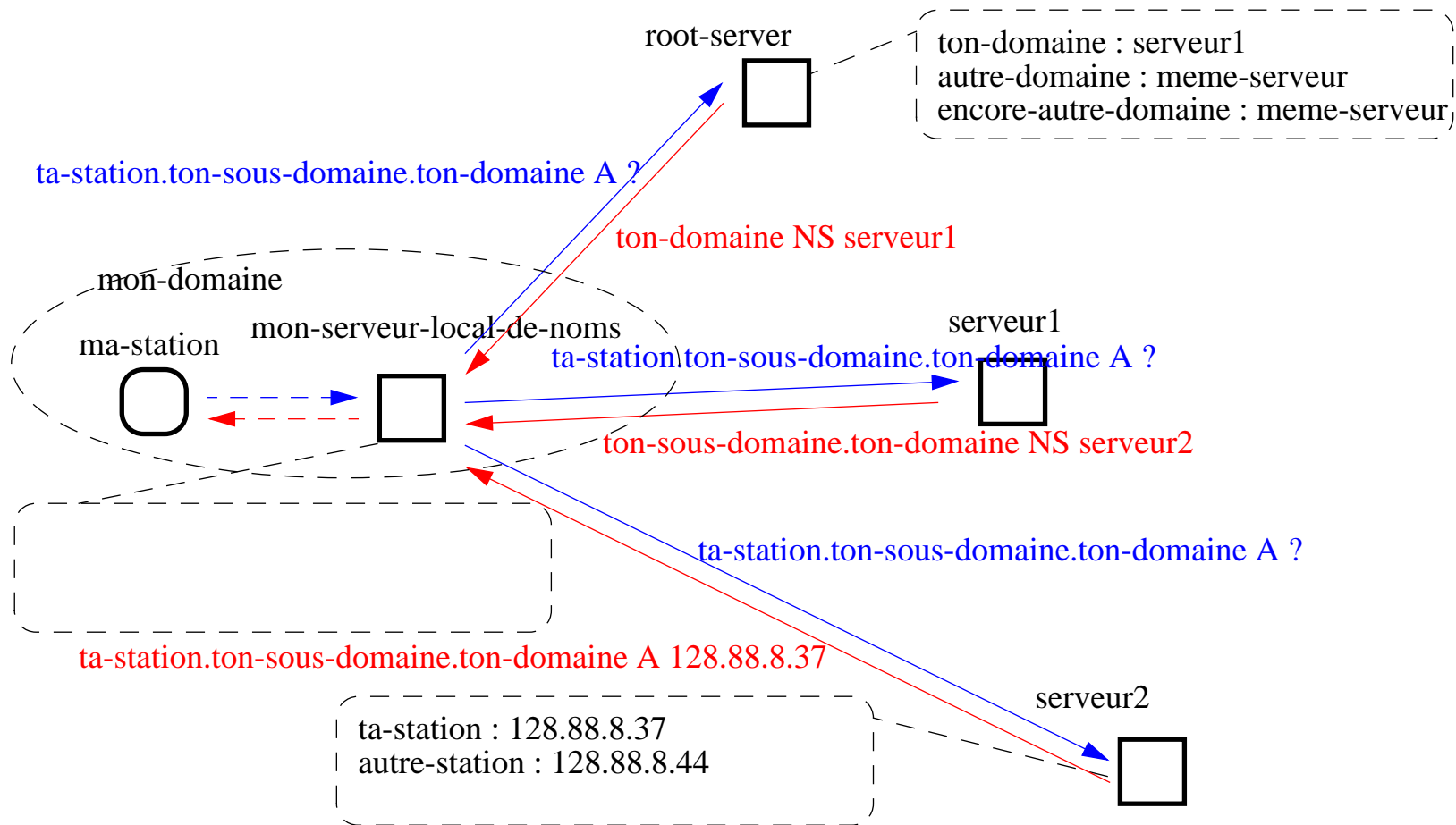


Des **cache**s peuvent être mis en oeuvre afin d'accélérer la résolution des noms :

- la dissémination des infos est effectuée en fonction de l'utilisation réelle
- les serveurs gérant ces caches stockent des infos qui peuvent ne pas concerner le domaine local.
- les informations obtenues peuvent être incorrectes ou incomplètes.

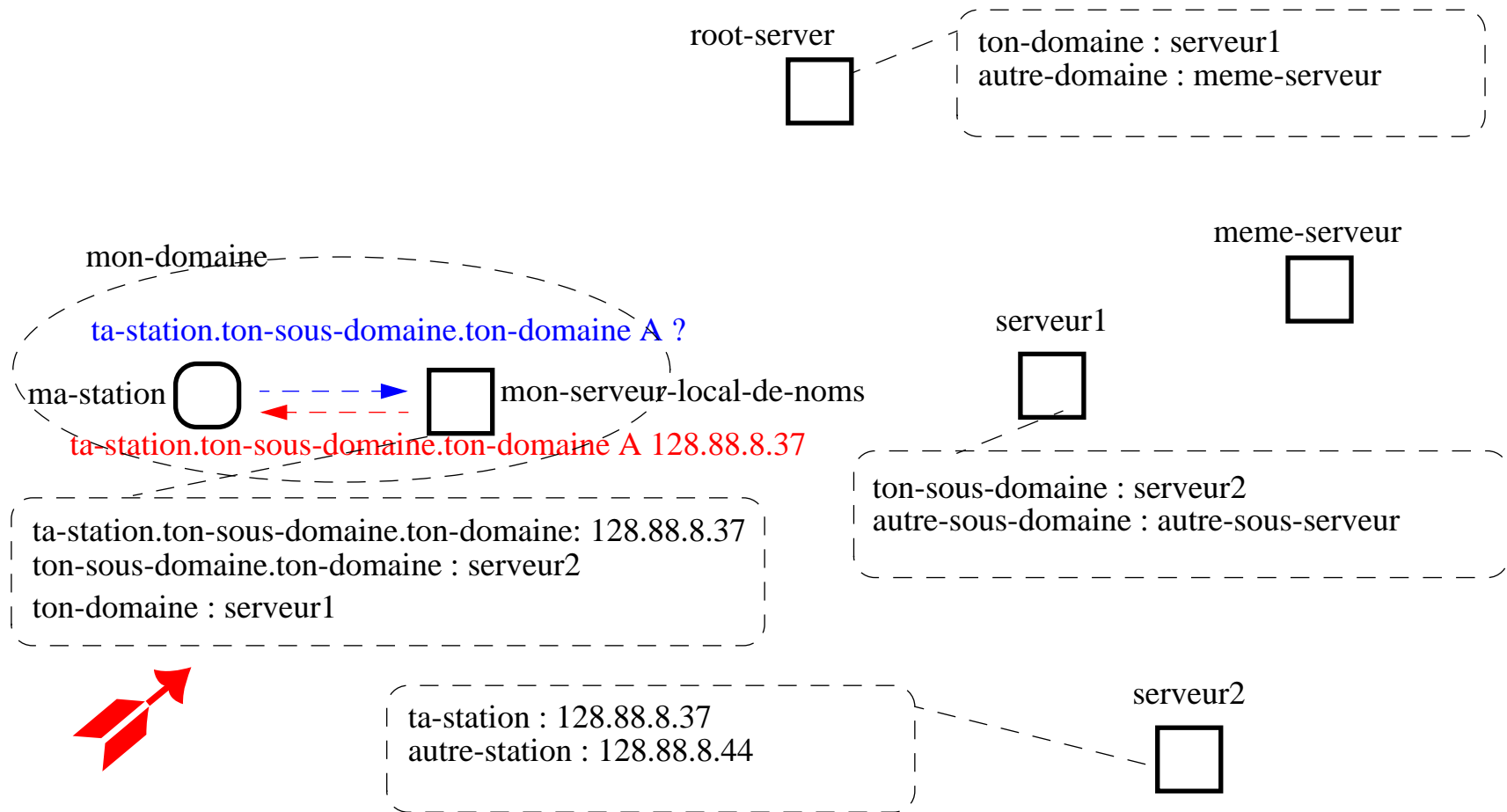
Remarque : un même serveur peut administrer plusieurs domaines de noms.

### 3.2.2 Exemple de résolution de noms



Cet enchainement est non-récuratif. Dans le cas récuratif, le serveur distant interroge les autres serveurs distants, puis transmet la réponse au serveur local. Cela surcharge les serveurs distants.

Le “cache”:



La réponse obtenue ne fait pas autorité : “non-authoritative”

### 3.3. Caractéristiques

Différences avec d'autres systèmes de résolution de noms :

- Les informations ne sont pas centralisées dans un seul fichier ou une station.  
    ☞ répartition
- Les informations obtenues peuvent être détenues par d'autres serveurs.  
    ☞ souplesse, fiabilité
- Les informations sont obtenues seulement quand on en a besoin  
    ☞ minimisation
- Le système gère un cache des informations obtenues précédemment  
    ☞ optimisation

Le DNS peut s'adapter :

- à différents réseaux de communication (“Class”)
- à différents systèmes d'exploitation
- à différents objets et informations (“Type”)  
    station, serveur de courrier, résolution inverse, informations diverses, etc.



## 4. Le résolution de noms sous Unix

### 4.1. Présentation

L'implémentation la plus courante sous Unix :

- BIND (“Berkeley Internet Name Domain”)
  - . Un “resolver”
  - . Des serveurs de résolution de noms
- Protocole normalisé :
  - . RFC 1034 : concepts (1987)
  - . RFC 1035: spécifications et implémentations

Le “[resolver](#)” :

- code qui effectue la demande,
- sous la forme d'une bibliothèque de fonctions : `gethostbyname()`, `gethostbyaddr()`
- s'exécute sur la station locale,
- présent dans toutes les stations.
- utilise
  - . soit des informations locales : `/etc/hosts`,

- soit des serveurs de noms distants : /etc/resolv.conf

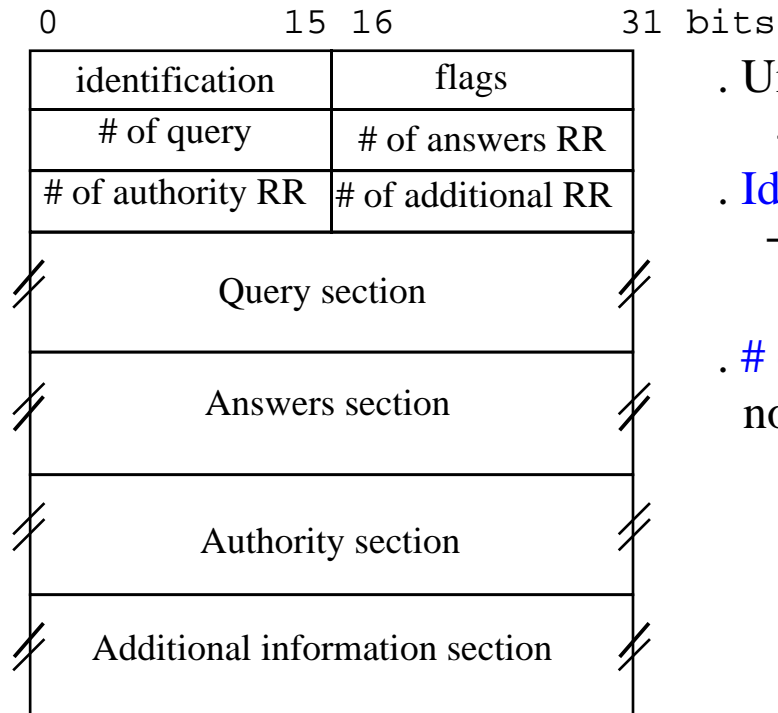
```
domain ifsic.univ-rennes1.fr
search ifsic.univ-rennes1.fr irisa.fr univ-rennes1.fr
nameserver 148.60.4.1
nameserver 148.60.4.5
nameserver 131.254.254.2
nameserver 129.20.128.2
```

### Le [serveur de noms](#) :

- répond aux demandes,
- procédé réparti de résolution des noms,
- s'exécute dans des stations distantes,
- un processus par serveur :
  - nommé : named = "name daemon", ou in.named,
  - associé au port n° 53 par TCP ou UDP,
  - présent sur tous les serveurs de noms.

 Une commande : nslookup

## 4.2. Format général des messages DNS



. Un entête de longueur fixe :

- 12 octets

. **Identification** :

- identifie le message : choisi par le demandeur, retourné par le répondeur.

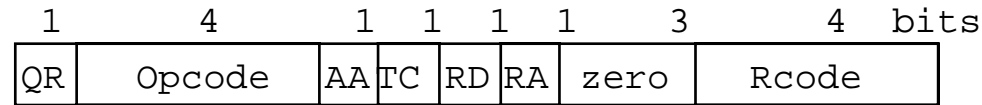
. **# of query** (resp. answers, authority, additional) :

nombre d'enregistrements dans la section correspondante.

- dans chaque section il peut y avoir 0 ou plusieurs enregistrements

- chaque enregistrement donnent des informations sur une ressource

### 4.3. Le champ **Flags**



**QR** (1bit) [Query/Response] : 0 = demande; 1 = réponse

**Opcode** (4 bits) : 0 = recherche standard; 1 = recherche inverse; 2 = demande de statut du serveur

**AA** (1bit) [Authoritative response] : le serveur de noms est celui qui a autorité sur le domaine .

**TC** (1bit) [Truncated] : Le réponse totale est plus longue (>512 octets) que celle contenue dans le message

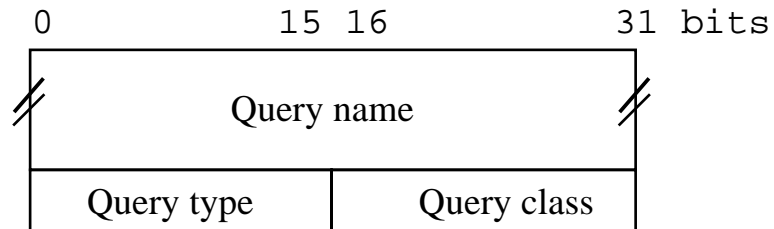
**RD** (1bit) [Recursion desired] : le demandeur demande au serveur de noms d'effectuer, si nécessaire, une recherche récursive. Ce serveur va s'occuper des recherches auprès d'autres serveurs. Sinon la recherche est itérative : le demandeur devra lui-même interroger les serveurs dont le nom figurera dans la réponse.

**RA** (1bit) [Recursion available] : le serveur indique au demandeur qu'il propose les services de recherche récursive.

**Zero** (3bits) : inutiles.

**Rcode** (4 bits) : code de retour, [0 = pas d'erreur, 1= erreur de format ; 2 = panne de serveur, 3 = le nom n'existe pas (fourni par un serveur ayant l'autorité), 4 = le serveur ne peut pas répondre, 5 = le serveur refuse de répondre].

#### 4.4. Un enregistrement de la section Questions



#### Query name :

- . Le nom de la ressource sur laquelle on veut obtenir des informations
- . Ce champ est de longueur variable

#### Query type :

- . Type de l'information recherchée
- . Autres enregistrements de réponse (champ Type).

Type	Code	Description
A	1	host address (nom → IP address)
NS	2	name server
CNAME	5	canonical name (alias)
SOA	6	start of authority (informations sur le domaine des noms)
PTR	12	pointer (adresse → nom)
HINFO	13	host info (informations diverses sur la station)
MX	15	mail exchanger (serveur de messagerie)
AAAA	28	IPv6 address (nom → IPv6 address)
ANY	255	all records req

#### Query class :

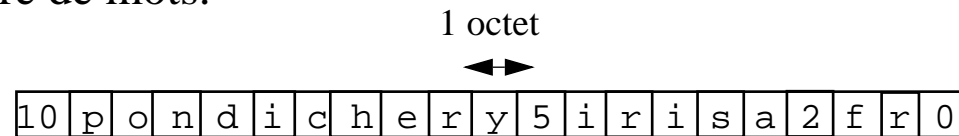
- . Famille protocolaire employée : Internet =1
- . Généricité

## 4.5. Le format des noms

Un **nom** : une suite de labels.

- . Chaque label étant une suite de caractères préfixée par un octet indiquant la longueur de la chaîne de caractères [1-63].
- . Terminé par un label nul (un octet à zéro) symbolisant la racine.
- . Peut ne pas se terminer en frontière de mots.

Par exemple : pondichery.iris.fr

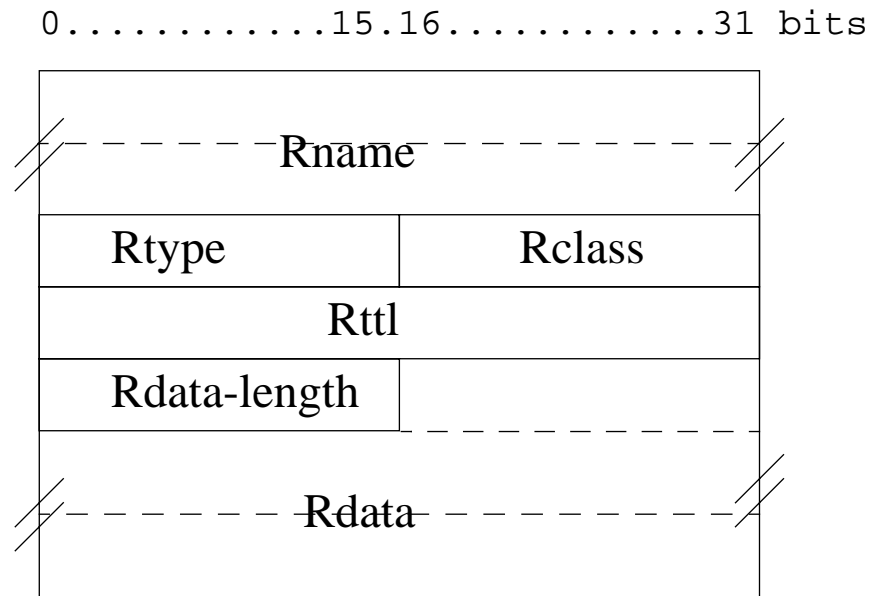


Autre exemple possible : 13.61.254.131.in-addr.arpa

Une représentation compressée existe :

- permet de ne pas représenter plusieurs fois la même suite de labels
- le premier octet possède les 2 bits de poids fort à 1
- les bits de poids faibles indiquent l'emplacement de la suite (son déplacement par rapport au début du message DNS)

## 4.6. Le format général des autres enregistrements

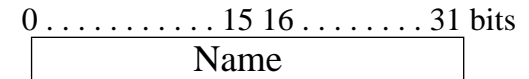


- **Rname** : le nom de la ressource dont on a obtenu l'enregistrement (cf. Qname)
- **Rtype** : Le type de la ressource (cf. Qtype)
- **Rclass** : la classe protocolaire (cf. Qclass)
- **Rttl** : la durée de vie de l'enregistrement
- **Rdata-length** : la longueur (en octets) du champ Rdata
- **Rdata** : la valeur de la ressource (dépend du type)

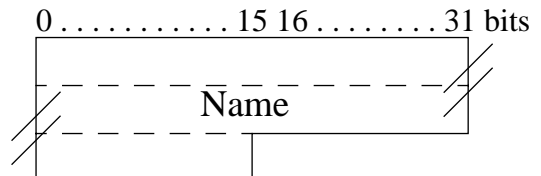
## 4.7. Différents types de ressources

Chaque type de ressource a un format particulier pour son champ Rdata :

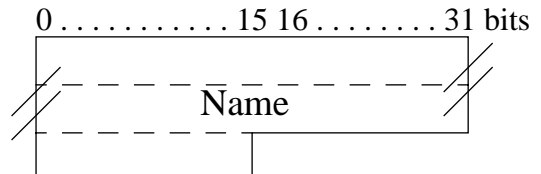
Type A [1] : adresse de la station de nom Rname  
 . sous Internet : adresse IP 32 bits



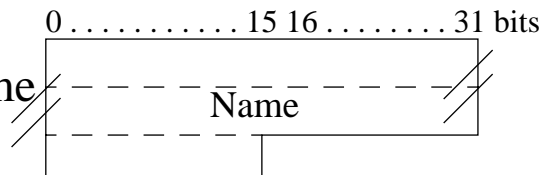
Type NS [2] : nom du serveur du domaine Rname



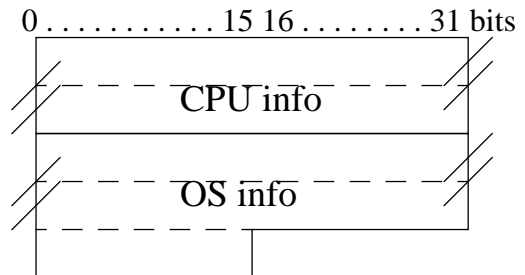
Type CNAME [5] : autre nom de la station Rname



Type PTR [12] : nom de la station dont l'adresse est Rname

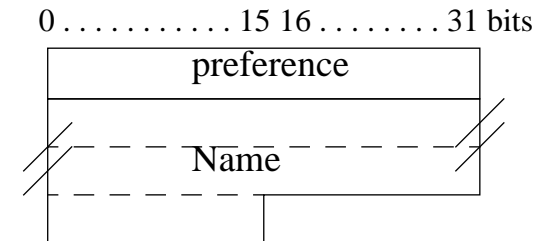


Type HINFO[13] : informations sur l'équipement Rname

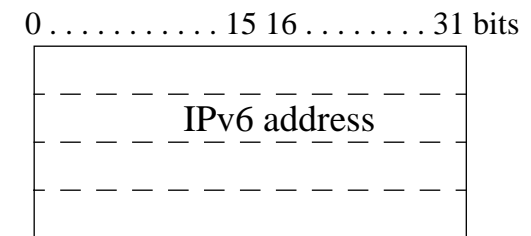




Type MX [15] : nom du serveur de messagerie pour le domaine Rname



Type AAAA[28] : adresse de la station de nom Rname  
 . pour IPv6 (16 octets)



Type \* [255] : toutes les informations disponibles sur Rname

## 4.8. Exemple

Tableau 1 : un message DNS

0	ce75	8180	0001	0002	0001	0000	0366	7470
16	0561	7070	6c65	0363	6f6d	0000	0100	01c0
32	0c00	0500	0100	0185	b900	170b	6272	6963
48	2d61	2d62	7261	6305	6170	706c	6503	636f
64	6d00	c02b	0001	0001	0001	85b9	0004	822b
80	0203	c037	0002	0001	0000	37ae	0012	0352
96	5330	0849	4e54	4552	4e49	4303	4e45	5400

- Entête du message DNS
  - . Identificateur du message : 0xce75,
  - . Flags : response, recherche standard, réponse d'un cache, message entier, récursion autorisée, récursion disponible, pas d'erreur (0x8180)
  - . 1 questions (0x0001), 2 réponses (0x0002), 1 serveurs (0x0001), 0 informations additionnelles (0x0000)
- Le premier enregistrement
  - . nom de la machine : ftp.apple.com
  - . type de la ressource : A = adresse IP (0x0001)
  - . classe protocolaire : Internet (0x0001)

## 5. Conclusion

Service de résolution de noms :

- . mais pas uniquement de noms: inverse, serveur de messagerie, infos, etc.
- . pour Unix et Internet mais pas uniquement.

Fiable, stable, facile à administrer et performant

On distingue les fonctions :

- . administration des noms, délégation
- . résolution des noms

Une arborescence de domaines, les noms sont une suite de labels.

Le procédé de résolution :

- . locale
- . par serveurs (primaire, secondaire, cache)

DNS : “Domain Name System”, BIND, “resolver”/”name server”