

---

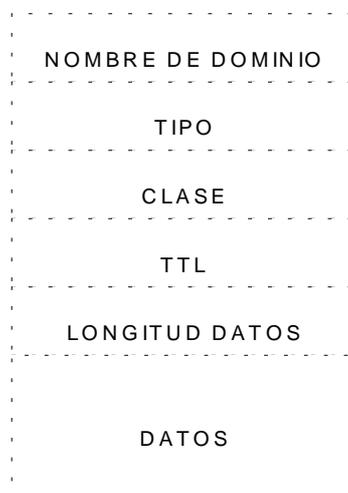
# Capítulo 7

## ESTRUCTURAS DE DATOS

Este capítulo describe el formato de mensaje que define el protocolo ANS, así como los denominados *registros fuentes* (estructuras de datos que utiliza el sistema para el almacenamiento e intercambio de información referente al espacio de nombres de dominio).

### REGISTROS FUENTES

La información de un sistema ANS se contiene dentro del denominado “*espacio de nombres de dominio*”. El espacio de nombres de dominio se puede asimilar a una estructura en árbol, en el que cada nodo se identifica con un nombre de dominio, asociado a cada nombre de dominio (o nodo del árbol) existe información que puede ser de diverso tipo, y a cada una de estas unidades de información se le denomina *registro fuente*. Un *registro fuente* puede ser definido con la siguiente estructura:



- 
- **NOMBRE DE DOMINIO:** Nombre de dominio al que pertenece el registro fuente.
  - **TIPO:** Tipo del registro fuente. Puede tomar los siguientes valores, y en función del valor variará el formato del campo de datos:
    - **NS(*Name Server*):** El campo DATOS del *registro fuente* contiene un servidor de nombres que sirve la zona del espacio de nombres del nodo al que pertenece el registro fuente.
    - **CNAME(*Canonical Name*):** El campo DATOS del *registro fuente* contiene un alias del nombre de dominio al que pertenece el registro fuente.
    - **SOA:** El campo DATOS del *registro fuente* contiene parámetros de mantenimiento de la zona a la que pertenece el registro fuente.
    - **PTR(*Pointer*):** El campo DATOS del *registro fuente* contiene información acerca del nombre de dominio que se corresponde con la dirección de red dada por el registro fuente en formato ATMA.INT.
  - **ATMA:** El campo DATOS del *registro fuente* contiene la dirección de red ATM de la máquina correspondiente al nombre de dominio del nodo al que pertenece el registro fuente.
  - **CLASE:** Clase del registro fuente. En el caso del sistema ANS siempre tomará el valor de clase IN(*Internet*).
  - **LONGITUD DE DATOS:** Longitud en bytes del campo de datos.
  - **DATOS:** Son los datos del registro fuente propiamente dicho. Su formato y/o contenido variará en función del tipo de registro fuente.

Estos registros fuentes son utilizados por el sistema ANS tanto para el almacenamiento de información referente a nombres de dominio, como también para el intercambio de dicha información entre los distintos elementos del sistema ANS.

---

## MENSAJE ANS

En el sistema ANS se define un formato MENSAJE que ha de ser el utilizado para cualquier comunicación entre dos entidades pares del protocolo ANS. Este mensaje está definido dentro del protocolo ANS y lo forman los siguientes campos:

CABECERA	
PETICION	la petición para el servidor de nombres
RESPUESTA	RRs contestando la petición
AUTORIDAD	RRs apuntando a otros servidores autoritarios
ADICIONAL	RRS con información adicional

- **CABECERA:** Formada por una serie de campos de longitud fija. Estos campos los utiliza el protocolo ANS para su funcionamiento interno.
- **PETICIÓN:** Formada por una lista de peticiones. Su número es variable. Los campos que forman una petición son los siguientes:
  - **NOMBRE DE DOMINIO:** Nombre de dominio por el que se realiza la petición.
  - **TIPO:** Tipo de datos por el que se realiza la petición.
  - **CLASE:** Clase de la petición. En el sistema ANS tomará el valor IN(*Internet*).
- **RESPUESTA:** Formada por una lista de registros fuentes que dan respuesta a una determinada petición. Su número es variable.
- **AUTORIDAD:** Formada por una lista de registros fuentes que dan información acerca de los servidores de nombres válidos para determinadas zonas. Esta información sirve de ayuda a la hora de cursar nuevas peticiones para obtener una respuesta válida. Su número es variable.
- **ADICIONAL:** Formada por una lista de registros fuentes que dan información *adicional* sobre nombres de dominio que aparecen en el mensaje. Su número es variable.

---

En el **ANEXO II** de esta memoria se describen los campos que forman el mensaje, así como los posibles valores que pueden tomar y su significado.

Además del formato de mensaje, el protocolo también define el orden de transmisión de los bits a la hora de mandar el mensaje a través de la red ATM. Las reglas utilizadas para dicho orden de transmisión de bits son muy simples:

1. Si el campo es multiocteto, se transmite primero el octeto más significativo.
2. Al transmitir un byte, primero se transmite el bit más significativo.

Existe una utilidad en el protocolo ANS y es la de compresión de mensajes. Con ayuda de la compresión se puede disminuir en gran medida el número de bytes a transmitir a la hora de enviar un mensaje a través de la red. La base del funcionamiento de la compresión y un ejemplo de la misma se muestra en el **ANEXO III** de esta memoria.

## APLICACIÓN AL SISTEMA ANS

El sistema ANS utiliza tres estructuras básicas de datos que son:

- MENSAJE ANS
- REGISTRO FUENTE
- PETICIÓN

Para facilitar el manejo de los datos se crea una clase para cada tipo de estructura de datos. Estas clases ofrecen una serie de métodos que implementan las posibles operaciones que podamos realizar con los datos, consiguiendo más simplicidad en el manejo de los datos. Las operaciones básicas que ofrecen estas clases son:

- **lee\_datos():** Obtiene los datos de un flujo de entrada(fichero ó memoria caché). Los datos leídos se almacenan en los atributos de la clase.
- **escribe\_datos():** Inserta los datos de los atributos en un flujo de salida(fichero ó memoria caché).
- **operator =( ):** Operador de igualdad. Copia los datos de dos clases.
- **operator ==( ):** Operador de comparación. Compara los datos de dos clases.
- **analiza\_mensaje():** Pasa los datos desde formato red a los atributos de la clase.
- **rellena\_mensaje():** Pasa los datos desde los atributos de la clase a formato red.

---

Además de estos métodos, existen otros que no son propios de cada clase sino que se implementan como función genérica y realizan las tareas de compresión y descompresión de los mensajes. Estas funciones son:

- **comprime():** Comprime dentro de un mensaje ANS el nombre de dominio que se le pasa como parámetro.
- **label\_to\_string():** Obtiene de un mensaje en formato red un nombre de dominio, esté o no comprimido.

Mediante la implementación de las clases anteriores, el manejo de los datos en el sistema ANS se simplifica en gran medida, de forma que pasa a ser transparente el formato de los datos que se manejen.

Un punto que cabe destacar a la hora del desarrollo del sistema ANS, es que con el uso de funciones virtuales en C++, ha sido posible el hacer transparente el formato de las estructuras de datos para el usuario de los objetos descritos anteriormente.