

---

# Capítulo 6

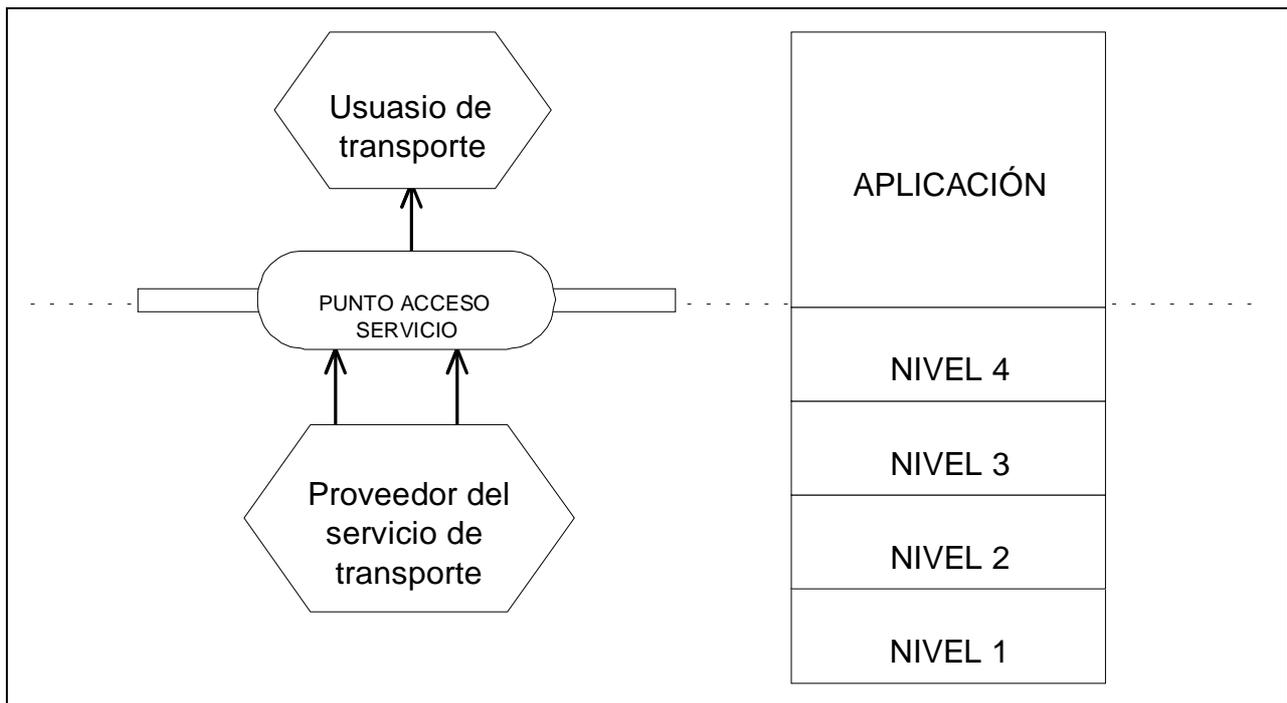
## COMUNICACIÓN EN LA RED ATM

---

En este capítulo se describe la librería desarrollada para la transmisión de datos entre de los distintos elementos del sistema ANS. Esta librería se basa en la librería de comunicación XTI(X Open Transport protocol) que ofrece un servicio de transporte a través de la red ATM.

### INTRODUCCIÓN

El sistema ANS se ha implementado para dar servicio directorio de nombres dentro de una red ATM. La transferencia de datos a través de la red se realiza siguiendo el protocolo ATM. ATM es un protocolo de transferencia de datos de nivel 2 en la torre de protocolos de capa OSI. Para poder hacer un uso eficaz de la red ATM, el sistema ANS utiliza la interfaz XTI(X Open Transport Protocol), que ofrece los servicios de la capa de transporte en forma de una serie de funciones implementadas dentro de una librería.



Los servicios básicos que ofrece esta librería al usuario del servicio de transporte son:

- Establecimiento de conexión de transporte.
- Envío de datos.
- Recepción de datos.
- Liberación de conexión.

Tal y como especifica la recomendación del protocolo ANS, sobre el protocolo ATM, se utilizará Capa de Adaptación ATM de nivel 5(AAL\_5) que ofrece servicios de segmentación, reensamblado y convergencia. Las conexiones utilizadas por el sistema ANS entre los distintos elementos de red serán de tipo circuito virtual conmutado(SVC), y serán señalizados a través del protocolo de señalización UNI de la red ATM.

---

## XTI.

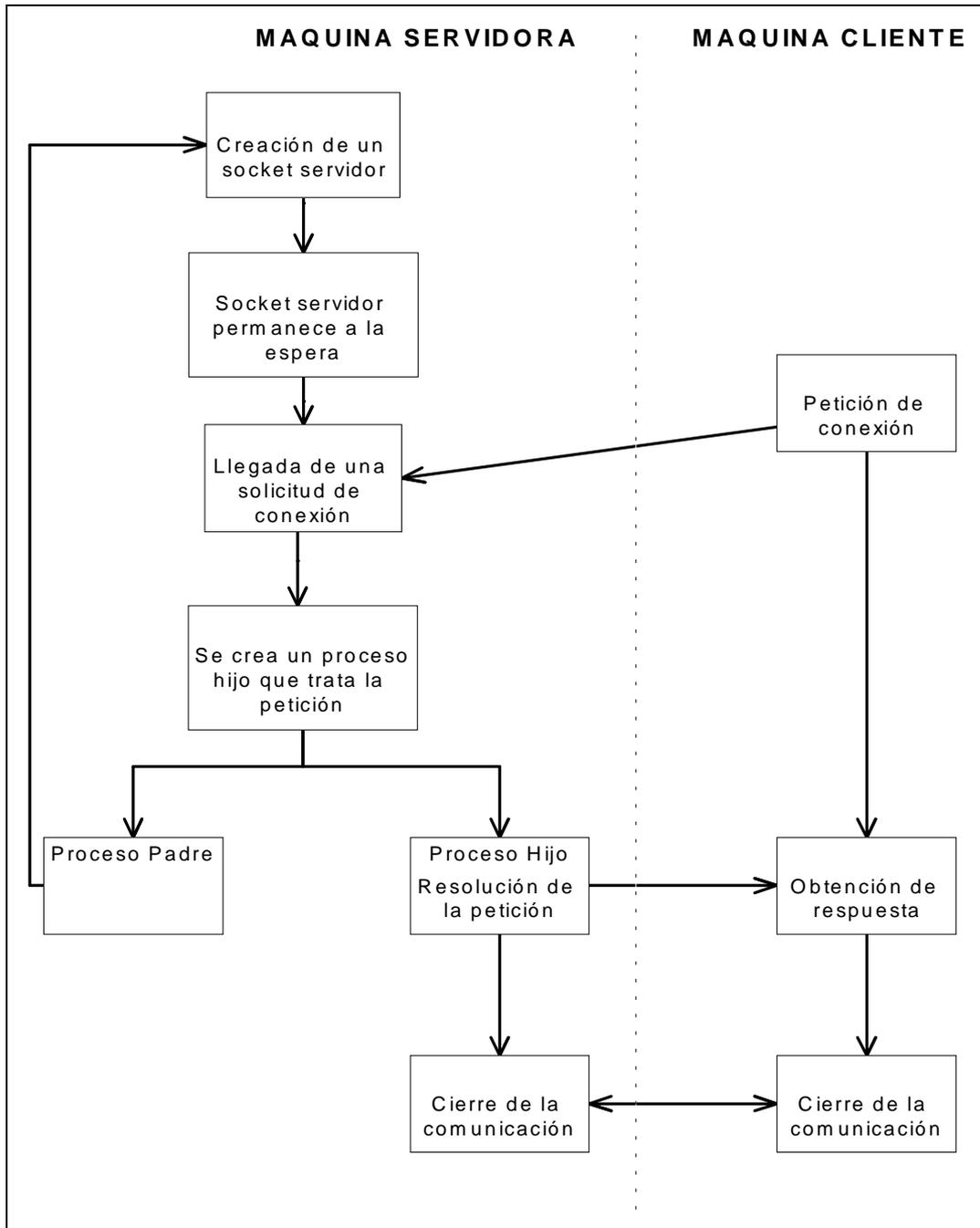
La librería XTI ofrece una serie de funciones y estructuras de datos que proveen de servicio de transporte para la red ATM. Las más significativas se listan a continuación:

- **t\_accept():** Acepta una solicitud de conexión entrante.
- **t\_bind():** Asocia una determinada dirección a un extremo de transporte.
- **t\_close():** Cierra un extremo de transporte.
- **t\_connect():** Establece una conexión entre dos extremos de transporte.
- **t\_listen():** Un extremo de transporte permanece a la espera de una solicitud de conexión por parte de otro extremo de transporte.
- **t\_open():** Establece un extremo de transporte.
- **t\_rcv():** Recibe datos de una determinada conexión.
- **t\_rcvconnect():** Recibe confirmación de conexión.
- **t\_rcvdis():** Recibe desconexión.
- **t\_snd():** Envío de datos a través de una conexión.
- **t\_snddis():** Envía una solicitud de desconexión.
- **t\_unbind():** Desliga la dirección asociada al extremo de transporte.

La interfaz XTI además ofrece otra serie de rutinas no listadas anteriormente y que dan otra serie de servicios que son también implementados normalmente por una capa de transporte. Estos servicios van desde el envío de datos urgentes, a la obtención de información acerca de una determinada conexión. Estos servicios no son utilizados por el sistema ANS por lo que no se tratan en este capítulo.

En el sistema ANS se utiliza la “*filosofía*” de comunicación por *sockets* en lo que respecta a la comunicación entre las distintas máquinas que forman el sistema. La programación con *sockets* se basa en establecer un puerto o punto de escucha en una máquina(máquina servidora) conectada a la red, que permanece a la espera de una petición de conexión por parte de otro extremo remoto(máquina cliente). Una vez llegada la solicitud de conexión, ésta pasa a ser establecida. Otra particularidad de la programación con *sockets* es que cada extremo de una conexión y/o cada puerto de escucha se identifica con un descriptor(número entero positivo), siendo único dentro de cada máquina.

En todas las conexiones que se establecen existirá una parte cliente que será la que solicite la conexión, y otra parte servidora que será la que permanece a la espera de una conexión. El funcionamiento normal del sistema de comunicación con sockets es el siguiente:



- 
1. En la máquina servidora se creará un socket servidor, por el que se permanece a la escucha en espera de una solicitud de conexión por parte de un posible cliente.
  2. Cuando entra una solicitud de conexión a la máquina servidora, se acepta dicha conexión, quedando establecida.
  3. Se crea un proceso hijo que será el que dialogue con la máquina cliente a través de la conexión establecida. El proceso padre vuelve a crear un nuevo socket servidor volviendo al paso 1 del algoritmo(espera de una petición de conexión).
  4. Una vez resuelta la petición de la máquina cliente por parte de la máquina servidora, se pasa a liberar la conexión establecida.

## APLICACIÓN AL SISTEMA ANS

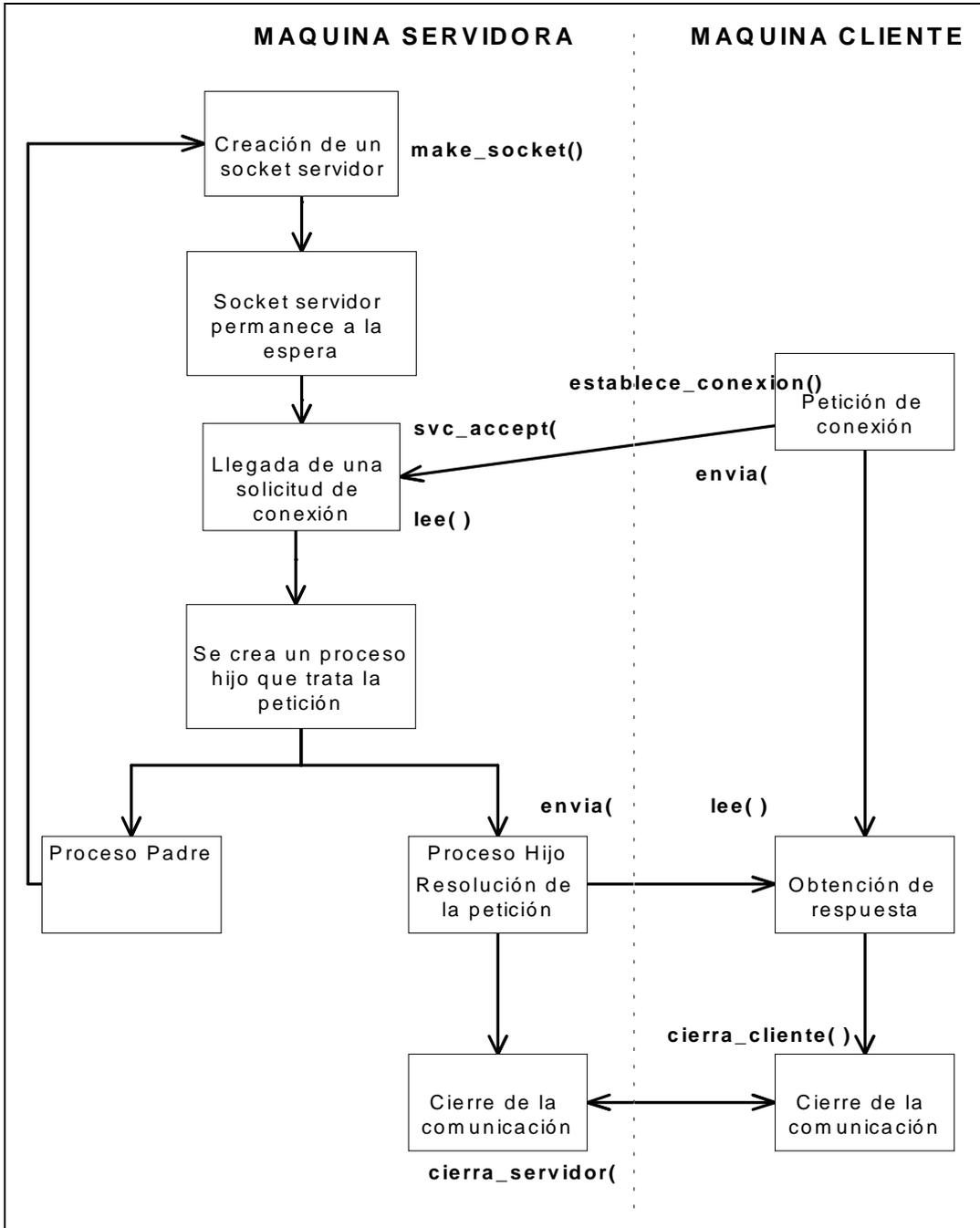
En el sistema ANS las conexiones que se utilizan se crean a través de Circuitos Virtuales Conmutados(SVCs) y el direccionamiento de las distintas máquinas se realiza a través de direcciones tipo AESA(ATM End System), ya que éstas son las únicas que permite la librería XTI.

Por otra parte el sistema ANS utiliza la filosofía de comunicación a través de sockets(modelo cliente-servidor). Para que sea más ameno el trato con las rutinas ofrecidas por XTI se crea una nueva librería de comunicación que ofrece una serie de funciones básicas del sistema de comunicación por sockets:

- **make\_socket:** Creación de un nuevo socket servidor. Las funciones XTI utilizadas para implementar esta función serán:
  - t\_open(). Abre el dispositivo de la tarjeta de interfaz ATM, asociándole un descriptor.
  - t\_bind(). Liga al descriptor el punto de transporte local.
- **establece\_conexión:** Envío de solicitud de conexión.
  - t\_open(). Abre el dispositivo de la tarjeta de interfaz ATM, asociándole un descriptor.
  - t\_bind(). Liga al descriptor el punto de transporte local.
  - t\_connect(). Envía una solicitud de conexión al extremo de transporte remoto.
- **svc\_accept():** Permanece a la escucha en espera de una solicitud de conexión y acepta la solicitud de conexión entrante.
  - t\_listen(): Permanece a la escucha a la espera de una petición de conexión.
  - t\_accept(): Acepta una petición de conexión entrante.

- 
- **envia():** Envío de datos.
    - **t\_snd():** Envía datos a través de una conexión establecida.
  
  - **lee():** Recepción de datos.
    - **t\_rcv():** Recibe datos a través de una conexión establecida.
  
  - **cierra\_cliente():** Liberación ordenada de conexión por parte del cliente.
    - **t\_snddis():** Envía solicitud de desconexión.
    - **t\_unbind():** Desliga el punto de transporte local del descriptor.
    - **t\_close():** Cierra el descriptor.
  
  - **cierra\_servidor():** Liberación ordenada de conexión por parte del servidor.
    - **t\_rcvdis():** Recibe solicitud de desconexión.
    - **t\_unbind():** Desliga punto de transporte local del descriptor.
    - **t\_close():** Cierra el descriptor.

La ubicación de estas funciones *genéricas* dentro del diagrama de flujo que siguen las comunicaciones dentro del sistema ANS se muestra a continuación:



Para más información acerca de los parámetros de conexión que utilizan los elementos de red del sistema ANS a través de la red ATM referirse al documento *ATM Name System Specification Version 1.0(af-saa-0069.000)* del *ATM FORUM Technical Committee*, punto 7.