AppleTalk Data Stream Protocol (ADSP)

One of the significant new features of AppleTalk included as part of system software version 7.0 is the **AppleTalk DSP (ADSP)**, which provides a full-duplex data stream connection between two nodes in an AppleTalk internet. (**ADSP** was available as an extension under System 6.) Like the AppleTalk Session Protocol, Printer Access Protocol, and AppleTalk Transaction Protocol, **ADSP** uses the Datagram Delivery Protocol (DDP) to send its data over the internet. Therefore, even though **ADSP** appears to its clients to handle data as a stream of bytes, the data is actually transmitted and received by DDP in packets. **ADSP** takes advantage of this fact by including control and status information in the DDP packet header. You can use **The .DSP Driver** routines described in the section called, **.DSP Driver Routines** to control an **ADSP** connection. **The .DSP Driver** takes care of the implementation of **ADSP** for you.

Every **ADSP** connection is between two sockets in an AppleTalk internet. Each socket can maintain concurrent **ADSP** connections with several other sockets, but there can be only one **ADSP** connection between any two sockets at one time. When a pair of sockets establishes an **ADSP** connection, each socket client initializes and maintains a certain amount of control and state information that it uses for synchronizing communication with the other socket client and for error checking.

The combination of a socket and the **ADSP** information maintained by the socket client is referred to as a **connection end.** When two connection ends establish communication, the connection is considered an **open connection.** When both connection ends terminate the link and dispose of the connection information each maintains, the connection is considered a **closed connection.** If one connection end is established but the other connection end is unreachable or has disposed of its connection information, the connection is considered a **half-open connection.** No communication can occur over a half-open or closed connection. To prevent a half-open connection from tying up resources, **ADSP** automatically closes any half-open connection that cannot reestablish communication within 2 minutes.

Using ADSP

You can use **ADSP** to implement a data stream connection between any two sockets on an internet. (Note that although there can be only one **ADSP** connection between any two sockets, a single socket can maintain connections with several other sockets.)
Opening and Maintaining an ADSP Connection describes how to open, maintain, and close a connection between two sockets on an internet.
Making a Connection Listener describes how to establish and use a **connection listener**-that is, a connection end that waits passively to receive a connection request and then passes the connection request on to its client, the connection server. Finally,
Writing a Connection Routine describes how to write a routine (referred to as a *user routine*) that **ADSP** calls when your connection end receives an unsolicited connection event.

When you establish an **ADSP** connection end, you must allocate a nonrelocatable block of memory for, and provide a pointer to, a **connection control block (CCB)** data structure, which is used by **ADSP** to store state information about the connection end. You may read the fields in the **CCB** to obtain information about the connection end, but you are not allowed to write to any of the fields except one, the userFlags field. The **CCB** requires 242 bytes and is defined by the **TRCCB** data type.