Specifying a Target Address

When you create an Apple event, you must specify the address of the target. The target address identifies the particular application or process that you want to send the Apple event to. You can send Apple events to applications on the local machine or on remote computers on the network.

These are the descriptor types that identify the four methods of addressing an Apple event.

typeApplSignature	The application signature of the target
typeSessionID	The session ID of the target
typeTargetID	The target ID record of the target
typeProcessSerialNumber	The process serial number of the target

To address an Apple event to a target on a remote computer on the network, you must use either the typeSessionID or typeTargetID descriptor type.

If your application sends an Apple event to itself, it should address the Apple event using a process serial number. Use the kCurrentProcess constant to specify the process serial number of your application. This is the fastest way for your application to send an Apple event to itself.

You can use any of the four address types when sending an Apple event to another application on the local computer. To allow the user to choose the target of an Apple event, use the **PPCBrowser** function. The **PPCBrowser** function presents a standard user interface for choosing a target application, much as the **Standard File Package** provides a standard user interface for opening and saving files.

The **PPCBrowser** function returns information about the application the user chose in a target ID record.

The **Event Manager** accepts all four types of addresses. Your application can also use another address type, if it also provides a coercion handler that coerces the address type into one of the four address types that the **Apple Event Manager** recognizes.

You specify the address using an address descriptor record (a descriptor record of data type AEAddressDesc). You must create a descriptor record of this type and then supply the address descriptor record as a parameter to the **AECreateAppleEvent** function.

You can use the **AECreateDesc** function to add any of the four target addresses to an address descriptor record. The following program shows four possible ways to create an address, each using a different address type.

Creating a target address

```
// Assuming inclusion of <MacHeaders>

#include <AppleEvents.h>

void SetTargetAddresses (AEAddressDesc *targetAddress1,
                         AEAddressDesc *targetAddress2,
                         AEAddressDesc *targetAddress3,
```

```
                                AEAddressDesc *targetAddress4,
                                TargetID *toTargetID,
                                ProcessSerialNumber *thePSN,
                                OSType theSignature,
                                PPCSessRefNum theSessionRef);

void SetTargetAddresses(AEAddressDesc *targetAddress1,
                                AEAddressDesc *targetAddress2,
                                AEAddressDesc *targetAddress3,
                                AEAddressDesc *targetAddress4,
                                TargetID *toTargetID,
                                ProcessSerialNumber *thePSN,
                                OSType theSignature,
                                PPCSessRefNum theSessionRef)
{
    OSErr  myErr;

    myErr = AECreateDesc(typeTargetID, (Ptr) toTargetID,
                        sizeof (*toTargetID), targetAddress1);
    myErr = AECreateDesc(typeProcessSerialNumber, (Ptr) thePSN,
                        sizeof (*thePSN), targetAddress2);
    myErr = AECreateDesc(typeApplSignature, (Ptr) &theSignature,
                        sizeof(theSignature), targetAddress3);
    myErr = AECreateDesc(typeSessionID, (Ptr) &theSessionRef,
                        sizeof(theSessionRef), targetAddress4);

// add your own error checking
}
```

You specify the descriptor type for the address, a pointer to the buffer containing the address, and the size of the buffer to the **AECreateDesc** function to create an address descriptor record. The **AECreateDesc** function returns an address descriptor record with the specified characteristics.

After creating an address, you can specify the address in the **AECreateAppleEvent** function.

When you specify an address to the **AECreateAppleEvent** function, the **Apple Event Manager** stores the address in the keyAddressAttr attribute of the Apple event.

You can use the **PPCBrowser** function to create a target ID record. The following program shows how to use the information returned from the **PPCBrowser** function to create a target ID record. You can then use **AECreateDesc** to create the address descriptor record for an Apple event.

```
//  Specifying a target address in an Apple event by using the
// PPCBrowser function

// Assuming inclusion of <MacHeaders>

#include <AppleEvents.h>

OSErr GetTargetAddress (Str255 myPrompt, Str255 myAppStr,
```

```
                                PortInfoRec *myPortInfo,
                                AEAddressDesc *targetAddress,
                                TargetID *toTargetID);
                                void DoError (OSErr myErr);


OSErr GetTargetAddress (Str255 myPrompt, Str255 myAppStr,
                                PortInfoRec *myPortInfo,
                                AEAddressDesc *targetAddress,
                                TargetID *toTargetID)
{
    OSErr  myErr;

    // Use PPCBrowser to let user choose the target

    myErr = PPCBrowser(myPrompt, myAppStr, FALSE,
                &(toTargetID->location), myPortInfo, nil, "\p");
    if  (!myErr)
        DoError (myErr);
    else  {
        toTargetID->name = myPortInfo->name;

        // Create the descriptor record for the target address)
        myErr = AECreateDesc(typeTargetID, (Ptr) toTargetID,
                        sizeof(*toTargetID), targetAddress);

        if (!myErr)
            DoError(myErr);

    }

    // add your own error checking
    return myErr;

}
```