Getting Data Out of an Attribute

You can use the **AEGetAttributePtr** or **AEGetAttributeDesc** function to get the data out of the attributes of an Apple event.

You can get the data out of an attribute using the **AEGetAttributePtr** function. You specify the Apple event that contains the desired attribute, the keyword of the desired attribute, the descriptor type the function should use to return the data, a buffer to store the data, and the size of this buffer as parameters to the **AEGetAttributePtr** function. The **AEGetAttributePtr** function returns the descriptor type of the returned data and the actual size of the data, and it places the requested data in the specified buffer.

For example, this code gets the data out of the keyEventSourceAttr attribute of an Apple event:

```
AppleEvent    theAppleEvent;
DescType      returnedType;
short         sourceOfAE;
Size          actualSize;
OSErr         myErr;

myErr = AEGetAttributePtr(&theAppleEvent, keyEventSourceAttr,
                         typeShortInteger, &returnedType, (Ptr) &sourceOfAE,
                         sizeof(sourceOfAE), &actualSize);
```

The keyEventSourceAttr keyword specifies the attribute to get the data from. The typeShortInteger descriptor type specifies that the data should be returned as a short integer; the returnedType variable contains the actual descriptor type that is returned. You also must specify a buffer to hold the returned data and specify the size of this buffer. The **AEGetAttributePtr** function returns the actual size of the data returned in the *actualSize* variable. You can check this value to make sure you got all the data.

As with the **AEGetParamPtr** function, you can request that **AEGetAttributePtr** return the data using the descriptor type of the original data, or you can request that the **Apple Event Manager** coerce the data into a **descriptor type** that is different from the original.

In this example, the **AEGetAttributePtr** function returns the requested data in the *sourceOfAE* variable, and you can determine the source of the Apple event by examining this value.

The next example shows how to use the **AEGetAttributePtr** function to get data out of the keyMissedKeywordAttr attribute. After your handler extracts all known parameters from an Apple event, it should check whether the keyMissedKeywordAttr attribute exists. If it does, then your handler did not get all of the required parameters.

Note that if **AEGetAttributePtr** returns the errAEDescNotFound result code, then the keyMissedKeywordAttribute does not exist-which indicates that your application has extracted all of the required parameters. If **AEGetAttributePtr** returns noErr, then the keyMissedKeywordAttribute does exist-which indicates that your handler did not get all of the required parameters.

```
 myErr = AEGetAttributePtr(&theAppleEvent, keyMissedKeywordAttr,
                               typeWildCard, &returnedType, nil, 0,
                               &actualSize);
```

The data in the keyMissedKeywordAttr attribute contains the first required parameter, if any, that your handler didn't retrieve. If you want this data returned, specify a buffer to hold the data and specify the size of the buffer. Otherwise, as in this example, specify NIL as the buffer and 0 as the size of the buffer.