
Reading and Writing Edition Data

With the **Edition Manager** you can read or write data a few bytes at a time instead of putting data into one block as the **Scrap Manager** does. This model is similar to the data fork of a Macintosh file. You can read sequentially by setting the mark to 0 and repeatedly calling read, or you can jump to a specific offset by setting the mark there. The **Edition Manager** also adds the capability to stream multiple formats by keeping a separate mark for each format. This allows you to write a few bytes of one format and then write a few bytes of another format, and so forth.

Once you have opened the edition container for a particular publisher, you can begin writing data to the edition. Use the **WriteEdition** function to write publisher data to an edition.

`err = WriteEdition (whichEdition, whichFormat, buffPtr, buffLen);`

The **WriteEdition** function writes the specified format (beginning at the current mark for that format type) from the buffer pointed to by the *buffPtr* parameter up to *buffLen* bytes.

After you open the edition container for a subscriber and determine which formats to read, use the **ReadEdition** function to read edition data.

`err = ReadEdition (whichEdition, whichFormat, buffPtr, buffLen);`

The **ReadEdition** function reads the data with the specified format (*whichFormat*) from the edition into the buffer. The **ReadEdition** function begins reading at the current mark for that format and continues to read up to *buffLen* bytes. The actual number of bytes read is returned in the *buffLen* parameter. Once the *buffLen* parameter returns a value smaller than the value you have specified, there is no additional data to read, and the **ReadEdition** function returns a **noErr** result code.