Opening an Edition Container to Read Data

Before reading data from an edition, you must use the **OpenEdition** function. Your application should only use this function for a subscriber. Use this function to initiate the reading of data from an edition.

err = **OpenEdition** (*subscriberSectionH, refNum*);

As a precaution, you should retain the old data until the user can no longer undo. This allows you to undo changes if the user requests it.

Your application can supply a procedure such as DoReadEdition to read in data from the edition to a subscriber. When your application opens a document containing a subscriber that is set up to receive new editions automatically, the **Edition Manager** sends you a Section Read event if the edition has been updated. The Section Read event supplies the handle to the section that requires updating. The Listing below provides an example of reading data from an edition.

Choosing Which Edition Format to Read

After your application opens the edition container for a subscriber, it can look in the edition for formats that it understands. To accomplish this, use the **EditionHasFormat** function.

err = **EditionHasFormat** (*whichEdition, whichFormat, formatSize*);

The **EditionHasFormat** function returns the noTypeErr result code if a requested format is not available. If the requested format is available, this function returns the noErr result code, and the *formatSize* parameter contains the size of the data in the specified format or kFormatLengthUnknown (-1), which signifies that the size is unknown.

After your application opens the edition container and determines which formats it wants to read, call the **ReadEdition** function to read in the edition data.

After you have completed writing the edition data into the subscriber section, call the **CloseEdition** function to close the edition. See the description of **Reading and Writing a Section** for detailed information.

The Listing below illustrates how to read data from an edition. As described earlier, you must open the edition, determine which formats to read, use the **ReadEdition** function to read in data, and then use the **CloseEdition** function to close the edition. This listing shows how to read only text.

```
// Listing. Reading in edition data
// Assuming Inclusion of MacHeaders
#include <Editions.h>

// This is a sample declaration for the pointer to your document information.
typedef struct {
    short resForkRefNum;
    FSSpec fileSpec;
    SectionHandle  sectionH;
    short nextSectionID;
```

```
    } *MyDocumentInfoPtr;

// Prototype the function as follows prior to calling it.
void DoReadEdition(SectionHandle);


void DoReadEdition(SectionHandle theSubscriber)

{
   EditionRefNum eRefNum;
   OSErr openErr;
   OSErr readErr;
   OSErr closeErr;
   MyDocumentInfoPtr thisDocument;
   Handle textHandle;
   Size formatLen;

   // User defined function prototypes
   MyDocumentInfoPtr FindDocument(SectionHandle);
   void MyErrHandler(OSErr);
   Handle GetTextInSection(SectionHandle,MyDocumentInfoPtr);

    // Find out which document this section belongs to.
    // The FindDocument function accomplishes this.
    thisDocument = FindDocument(theSubscriber);

    // Open the edition for reading.
    openErr = OpenEdition(theSubscriber, &eRefNum);

    if ( openErr )
       // If the open failed, then most likely you can't read,
       // so don't continue with this operation.
        MyErrHandler(openErr);

   // Look for 'TEXT' format.
   if (EditionHasFormat(eRefNum, 'TEXT', &formatLen) == noErr) {
      // Get the handle of location to read to.
       // The GetTextInSection function accomplishes this.
       textHandle = GetTextInSection(theSubscriber, thisDocument);

       SetHandleSize(textHandle, formatLen);
       HLock(textHandle);
       readErr = ReadEdition(eRefNum, 'TEXT', *textHandle, &formatLen);
       HUnlock(textHandle);

       if (readErr == noErr) {
          // The read was successful; now close the edition.
           // When successful = TRUE, the section data = edition data.
           closeErr = CloseEdition(eRefNum, TRUE);
           return;
       }
   }
   // 'TEXT' format wasn't found or read error; just close
   // the edition. FALSE tells the Edition Manager that your application
   // did not get the latest edition.
   closeErr = CloseEdition(eRefNum, FALSE);
```

```
}
```