SectionRecord   structure

#include <Editions.h>

typedef struct **SectionRecord** {

		Size	Offset	Description
char	version;	1	0	always 0x01 in system 7.0
SectionType	kind;	1	1	stSubscriber or stPublisher
UpdateMode	mode;	2	2	auto or manual
TimeStamp	mdDate;	4	4	last change in document
long	sectionID;	4	8	app. specific, unique per document
long	refCon;	4	12	application specific
AliasHandle	alias;	4	16	handle to Alias Record
long	subPart;	4	20	which part of container file
struct SectionRecord	**nextSection;	4	24	for linked list of app's Sections
Handle	controlBlock;	4	28	used internally
EditionRefNum	refNum;	4	32	used internally
} **SectionRecord**;		36		

typedef SectionRecord ***SectionPtr**;
typedef SectionRecord ****SectionHandle**;

Field descriptions

version
: Indicates the version of the section record, currently 0x01.

kind
: Defines the section type as either publisher or subscriber with the stPublisher or stSubscriber constant.

mode
: Indicates if editions are updated automatically or manually.

mdDate
: Indicates which version (modification date) of the section's contents is contained within the publisher or subscriber. The mdDate is set to 0 when you create a new subscriber section, and is set to the current time when you create a new publisher. Be sure to update this field each time publisher data is modified. The section's modification date is compared to the edition's modification date to determine whether the section and the edition contain the same data. The section modification date is displayed in the publisher and subscriber options dialog boxes. See the section, **Closing an Edition** for detailed information.

sectionID
: Provides a unique number for each section within a document. A simple way to implement this is to create a counter for each document that is saved to disk with the document. The counter should start at 1. The section ID is currently used as a tie breaker in the GoToPublisher function when there are multiple publishers to the same edition in a single document. The section ID should

not be 0 or -1. See the section,
Duplicating Publishers and Subscribers for
information on multiple publishers.

refCon   Available for application-specific use.

alias   Contains a handle to the alias record for a particular
                          section within a document.

Whenever the user creates a publisher or subscriber, call the **NewSection**
function to create the section record and the alias record.

  err = **NewSection** (&container, &sectionDocument, kind, sectionID,
              initialMode, &sectionH);

The **NewSection** function creates a new section record (either publisher or
subscriber), indicates whether editions are updated automatically or
manually, sets the modification date, and creates an alias record from the
document containing the section to the edition container.

The *sectionDocument* parameter can be NIL if your current document has
never been saved. Use the **AssociateSection** function to update the
alias record of a registered section when the user names or renames a
document by choosing **Save As** from the **File** menu. If you are creating a
subscriber with the *initialMode* parameter set to receive new editions
automatically, your application receives a Section Read event each time a new
edition becomes available for this subscriber.

If an error is encountered, the *sectionH* parameter is set to NIL. If not,
sectionH contains the handle to the allocated section record.

Set the *initialMode* parameter to the update mode for each subscriber and
publisher created. You can specify the update mode using these constants:

   Constants

   sumAutomatic   subscriber receives new editions automatically
   sumManual   subscriber receives new editions manually
   pumOnSave   publisher sends new editions on save
   pumManual   publisher does not send new editions until user request

See **Using Publisher and Subscriber Options** for detailed information
on update modes for publishers and subscribers.