Using BookMaster on EVIEVM and EVIEMVS

Evie Cooper, EVIEVM(WEC)

Contents

EVIEVM Instructions

To format a document with BookMaster on EVIEVM, you will likely want to follow several steps:

1. Access the BookMaster, SCRIPT, LEXX, and artwork converter programs
2. Use **LEXX** to compose the BookMaster markup file
3. Prepare artwork and convert it into the correct format with converter programs found on the **PROGS1** minidisk
4. Use **SCRIPT** to generate the AFP or PostScript output
5. Converting PostScript to a PDF or previewing the AFP directly
6. Produce HTML output

Finding the Programs

If you have the **VMLINK** program (and you should unless you are under exceptional circumstances), you will be able to link the **PROGS1** minidisk:

```
 U1       VMLINK   S1  V 260  Trunc=260 Size=18 Line=1 Col=1 Alt=2
Cmd   Nickname Vdev Fm Ext Lm  Category  Description
      OVVM     =    = /        OFFICE    OfficeVision/VM
      OVHELP   =    = /        OFFICE    OfficeVision/VM Help
      TCPIP    =    = /        NETWORK   TCP/IP user programs
      VMWEBSRV =    = /        NETWORK   VM WebGateway OV/VM Stub
      C370     =    = /        SYSTEM    C/370 Library
      ISPF     =    = /        SYSTEM    ISPF/VM (Required for OV/VM)
      IPFVM    =    = /        SYSTEM    IPF User Menus (Requires ISPF)
*     PROGS1   =    = /        OFFICE    SCRIPT, TOOLS, DW/370, APL
      DB2VM    =    = /        DATABASE  Db2/VM user programs
      DB2REXX  =    = /        DATABASE  Db2/VM REXX SQL
      QMF      =    = /        DATABASE  Db2/VM frontend, run DSQ2EINV
      APPC     =    = /        NETWORK   APPC/VM user programs
      REXXC    =    = /        COMPILER  REXX compiler
      GAMES    DIR  = /        TOOLS     TOOLS: games VMTOOLS
      PUBDOCS  DIR  = /        TOOLS     TOOLS: documentation
      VMTOOLS  DIR  = /        TOOLS     TOOLS: VM tools
      WWCHAT   DIR  = /        TOOLS     TOOLS: chat
1= Help       2= Refresh  3= Quit      4= Sort(name)  5= Link        6= A
7= Backward   8= Forward  9= Category 10= Detach     11= Filelist   12= C
DMSVML2060I SCRIPT, TOOLS, DW/370, APL linked as 120 file mode Z
====>
```

Figure 1. VMLINK menu

Note that **LEXX** is also present there.

Using LEXX to Write Your Document

BookMaster markup input files are of the filetype **SCRIPT.**

This is not a guide on LEXX, but you will want to remember at least a few things:

- You will need to initially press PF2 to get some space to type in
- Upon pressing Enter, LEXX will format the current line and fill it
- You will want to press Enter within about 5 columns of the right vertical-bar margin
- LEXX automatically renders nontypable symbols (like brackets) in cyan

```
Document: MYDOC SCRIPT A1       At: ":docprof "..   19 changes


   :userdoc.

   :prolog.

     :docprof layout=1 style=mlnotes.

     :title.Using BookMaster on EVIEVM

     :author.WEC/EVIEVM
   :eprolog.

     :frontm.
     :tipage.

     :toc.

     :body.

 PF1  Help    PF2  Opencl  PF3  Quit    PF4  Copy    PF5  Move    PF6  ?
 PF7  Up      PF8  Down    PF9  =       PF10 Select  PF11 Splitj  PF12 F
```

Figure 2. LEXX editing a BookMaster file

Converting Artwork

If you wish to draw up pictures on the mainframe, invoke the **ADMDRAW** program (or the older **ADMDRAWO** program) to start a GDDM drawing program. Of course, this will require the usage of a terminal capable of both programmed symbol and vector graphics.[1]

Once you are done drawing, go to File and Save; save that file into a file of your name choice. It will assume the filetype of **ADMGDF**.

[1] Right now, this is *IBM Personal Communications* (aka PCOMM), *IBM Host On-Demand* (aka HOD), and *DN-Computing Quick3270*

To convert these, use either of the two scripts depending on the format you wish the image to be in:

MAKEEPS Encapsulated PostScript - *EPS*

MAKEPSEG 38PPN Page Printer Segment - *PSEG38PP*[2]

These programs accept **ADMGDF** files, a vector graphics image. The related **ADMIMAGE** file is a monochrome bitmap -- these are not as useful.

If you want to preview a GDF, run this to start OPS:

```
ADMOPSLA TEST1
```

The *TEST1* portion refers to a dummy presentation file of the same name.

Then, press PF1, then PF5 -- a dialog will come up that will let you view the images.

Workstation Artwork

If you have an old version of CorelDRAW, like version 3 or 4, you can import and export the **PIF** format. A *PIF* (Picture File) is a slightly-modified *ADMGDF* file, and is readily converted to and from *ADMGDF.*

On CorelDRAW, you need to make sure that that the image canvas is rather large; for a logo-style image that might look good on a wide-horizontal canvas, 70 inches by 40 inches would be ideal.

You will then need to convert the *PIF* (which CorelDRAW exports with the **PF** format) file to a *ADMGDF.* However, before you can do this, you need to upload the image to the mainframe with the correct attributes:

```
RECFM = F or V
LRECL = 80 for V, or 400 for F
```

Then, convert it:

```
ADMUPCV filename PIF A (PUT filename
```

You can then use the **MAKEEPS** or **MAKEPSEG** commands.

Including Artwork

BookMaster requires you, no matter if you are on MVS or VM, to specify the CMS file mode and/or MVS partitioned dataset that the image files can be found. While this is not a tutorial on BookMaster as a language, you would want to structure a document like this:

[2] Note that **PSEG38PP** files are monochrome!

```
:userdoc.
:prolog.
:title.My Cool Document
:author.Joe Blow
.nameeps cmstype=eps pdsfull='wec.doc.eps'
:&eprolog.
:frontm.
:tipage.
:body.
:h1.Some Art
:fig width=column place=inline frame=none.
:artwork name=mypic align=center.
:figcap.Some Cool Picture
:efig.
:euserdoc.
```

This includes the **MYPIC EPS** file if you are outputting PostScript; for AFP output, it will automatically locate **MYPIC PSEG38PP** and slipstream it in.

The **G 1** option in **BOOK4PS OPTIONS** and **BOOK4AFP OPTIONS** will automatically inline graphics; this will be described in detail soon.

Running SCRIPT

Since you have the **PROGS1** disk accessed, you can now SCRIPT in various ways with various options files:

```
SCRIPT myfile (OPT(options)
```

Your system programmer has placed several *OPTIONS* files for your aid:

BOOK4AFP	BookMaster 4, AFP
BOOK4PS	BookMaster 4, PostScript
BOOK3AFP	BookMaster 3, AFP
BOOK3PS	BookMaster 3, AFP
BOOK4LP	BookMaster 4, line printer carriage control
BOOK3AFP	BookMaster 3, line printer carriage control
BOOK4TRM	BookMaster 4, 3270 terminal

All AFP output is made with the **PSEG38PP** file mode. PostScript produces **LISTPS**.

Now, SCRIPT provides a *return code* that provides a one-shot look at any errors that you may have incurred:

0	No errors
4	Warning
8	Error, but output still produced
12	Fatal error, SCRIPT died

Error descriptions show up on your console. If you wish for these to be included in the resultant document for some reason, simply edit the SCRIPT options file's **MESSAGE** line to read *MESSAGE(DELAY,TRACE)*

Converting PostScript to PDF

First, the **:xmp** and **:screen** tags require a special font -- BookMaster Gothic. This is provided on the **TOOLS1** minidisk with files that are of the following two file patterns:

1. **EDF* PFA** - PostScript font data
2. **EDF* PFM** - PostScript font metrics

You will also want to get the IBM Logo files, such that you can format anything that uses the *IBMUGPL* or similar BookMaster styles (see the BookMaster manuals on what this means and how to create custom ones):

1. **IBMLOGO8 PFA**
2. **IBMLOGO8 PFM**

If you wish to download these files all-in-one, go here:[3]

`http://evievm.pubvm.org/eweb/bookfnt.tar`

You then need a program that can convert a PostScript file to a PDF:

1. Adobe Acrobat Distiller
2. Artifex GhostScript (open-source)

GhostScript Fonts

Deposit the fonts (either from the tar file or downloaded to your workstation) into the following location, provided that GhostScript is installed to the */usr/local* prefix:

`/usr/local/share/ghostscript/fonts`

You then need to edit a control file, where **VERS** is the version of GhostScript you have *(you can probably just tab-complete the path here):*

`/usr/local/share/ghostscript/VERS/Resource/Init/Fontmap.GS`

Next, enter the following at the bottom:

[3] Location subject to change!

```
/BookMasterGothic-Roman          (EDFBG.pfa) ;
/BookMasterGothic-Bold           (EDFBGB.pfa) ;
/BookMasterGothic-BoldItalic     (EDFBGBI.pfa) ;
/BookMasterGothic-Italic         (EDFBGI.pfa) ;
/BookMasterSymbol-Regular        (EDFBGM.pfa) ;
/BookMasterGothic-Reverse        (EDFBGR.pfa) ;
/BookMaster                      (EDFBL.pfa) ;
/BookMaster-Bold                 (EDFBLB.pfa) ;
/BookMaster-BoldItalic           (EDFBLBI.pfa) ;
/BookMaster-Italic               (EDFBLI.pfa) ;
/BookMasterReverse               (EDFBLR.pfa) ;
/BookMasterSpecials              (EDFBS.pfa) ;
/BookMasterSpecials-Bold         (EDFBSB.pfa) ;
/BookMasterSpecials-BoldItalic   (EDFBSBI.pfa) ;
/BookMasterSpecials-Italic       (EDFBSI.pfa) ;
/BookMasterSpecialsReverse       (EDFBSR.pfa) ;
```

After downloading the **LISTPS** file to your workstation, convert it to PDF:

```
$ ps2pdf mydoc.ps
```

Adobe Acrobat

Note that this requires Distiller -- this is is in the paid version of Distiller. Old versions of it (suitable for the generation of PC that EVIEVM matches) can be found on *winworldpc.com* and similar sites.

Extract or place the above font files into a directory of your choosing, and open Distiller. Go to the options dialog, and add a path for the files -- if you downloaded the tar file, ensure that you “flatten” it into a single directory (that is, move all of the files from the three subdirs in the archive into one directory).

You can then process the **LISTPS** file directly on Distiller, or, preferably, by picking the *New* option in Acrobat's *File* menu.

Viewing AFP Directly

There are two ways to do this. First, you can use the **SHOW** program on *VMTOOLS*:

```
SHOW myfile LIST38PP
```

Alternatively, to directly invoke the program that **SHOW** calls:

```
ADMUBCDV myfile LIST38PP A
```

If you are for some reason using TSO:

```
ADMUBCDT 'MY.AFP.FILE'
```

If you are on an old workstation, there is a program called **IBM AFP Workbench** for at least Windows NT and OS/2; the version for OS/2 can be found on the Misc CD of the OS/2 Warp 4 DEMOpkg. Versions for Windows can be found in various locations, including the installers for CA/400.[4]

[4] Client Access/400

Converting AFP to PostScript

For some reason, if you wish to convert AFP to PostScript, you can do so with **lp3820** (which is on the **PROGS1** minidisk).

Use it as follows:

```
GLOBAL TXTLIB EDCBASE IBMLIB CMSLIB
GLOBAL LOADLIB EDCLINK
LP3820 mydoc.LIST38PP
```

This depends on the **LP3820 PROFILE** file present on that disk!

This can also be used to convert *PSEG38PP* files to PostScript for viewing!

Producing HTML

Before you can produce HTML, you need to convert any GDF artwork with the **ADMUGIF** program:

```
ADMUGIF filename (W 1000 D 1000
```

Then, run **B2H** to convert it into HTML:

```
B2H mydoc SCRIPT
```

This results in the creation of **mydoc HTML**, which can then either be placed in your SFS root directory and shared with EnterpriseWeb/VM.[5] Please ensure you download the HTML file in *ASCII mode* and download any GIFs in *binary mode* if you are using.*IND$FILE.*

Some Additional File Conversions

ADMGDF to CGM:

```
ADMUGC filename ADMGDF A filename CGM A
```

PSEG3820 to ADMIMG:

1. Run **ADMIVU**
2. Go to option 2 (View Image)
3. For View Image, enter the filename
4. Upon hiting enter, you will need to enter 3 for the Selection field (a non-GDDM file)
5. Upon hitting enter again, enter **filename PSEG3820**for the File ID

ADMIMG to PSEG3820/LIST38PP:

1. Run **ADMIVU**
2. Load an image of the ADMIMG format

[5] Please see the document named **EVIEWEB SCRIPT** on the**PUBDOCS** TOOLS disk.

3. Go to Save Image, set the save type to non-GDDM files, set the type to whatever you want in the menu

CGM to ADMGDF:

```
ADMUCG filename CGM A filename
```

ADMGDF to PIF:

```
ADMUPCV filename PIF A (GET filename
```

EVIEMVS Instructions

I highly recommend that you use BookMaster on VM. BookMaster on MVS is made much more difficult by a lack of LEXX, no**LP3820**, and other things like that. Alas, I have tried to make EVIEMVS's BookMaster system as good as possible!

Finding the Programs

The following datasets contain things you need:

```
DCF140.EDFEXC1  - JCL samples for line-numbered documents
DCF140.EDFMAC1  - the actual BookMaster macros
DCF140.EDFPFA1  - .pfa PostScript fonts
DCF140.EDFPFB1  - .pfb PostScript fonts
DCF140.EDFPSF1  - .afm PostScript fonts (FONTPS library)
DCF140.EDFSAM1  - samples
DCF140.EDF38P1  - 38PPN fonts
DCF140.EDF3821  - 3820 fonts
DCF140.EDF4251  - 4250 fonts
DCF140.DCFIMAGE - 3800 fonts
DCF140.DCFLOAD  - the actual SCRIPT/VS programs
DCF140.MACLIB   - the actual GML macros
```

You need to ensure that your TSO logon procedure has the**DCF140.DCFLOAD** in the **STEPLIB** concatenation, like in the **DBSPROC** logon procedure:[6]

```
//STEPLIB DD DSN=GDDM.SADMMOD,DISP=SHR,UNIT=3390,VOL=SER=Z5RES2
//        DD DSN=DCF140.DCFLOAD,DISP=SHR
//        DD DSN=GDDM.SADMMOD,DISP=SHR,UNIT=3390,VOL=SER=Z5RES2
```

The **SCRIPT** command is provided by a CLIST baked into TSO, so there is no need to add anything to your TSO search order there.

Using ISPF to Write Your Document

Before creating any documents, you will want to allocate a dataset: *userid.SCRIPT* - RECFM=V, LRECL=132, PDS

Then, start creating a file in that dataset as shown above in the EVIEVM section.

Running SCRIPT

There is no forgiving way to do this, so, here goes:

[6] This is the default for EVIEMVS users.

```
SCRIPT infile
    PROFILE('DCF140.EDFMAC1(EDFPRF40)')
    LIB('DCF140.EDFMAC1')
    DEV(PSA)
    PSOUT(ASCII)
    FPASSES(3)
    INDEX
    MESSAGE(DELAY TRACE)
    PAGE(FROM * TO *)
    SYSVAR(D NB  H NO  X NO  Q 1  G INLINE)
    CONTINUE
    FONTLIB('DCF140.FONTPS')
    FILE(outfile)
```

This is less than ideal, since you will actually run out of room on either your ISPF command line or TSO command field, so it is best to **run SCRIPT in batch.**

PostScript

You can run this from a batch job with **IKJFET01**, like in this example:

```
//WECSCRPT JOB (1),CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=WEC
//STEP1 EXEC PGM=IKJEFT01
//STEPLIB DD DSN=DCF140.DCFLOAD,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
 SCRIPT 'WEC.DOC.SCRIPT(DEMO)'               -
     PROFILE('DCF140.EDFMAC1(EDFPRF40)')     -
     LIB('DCF140.EDFMAC1')                   -
     DEV(PSA)                                -
     PSOUT(ASCII)                            -
     FPASSES(3)                              -
     INDEX                                   -
     MESSAGE(DELAY TRACE)                    -
     PAGE(FROM * TO *)                       -
     SYSVAR(D NB  H NO  X NO  Q 1  G INLINE) -
     CONTINUE                                -
     FONTLIB('DCF140.FONTPS')                -
     FILE('WEC.SCRIPT.LISTPS')
/*
```

AFP

```
//WECSCRPT JOB (1),CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),NOTIFY=WEC
//STEP1 EXEC PGM=IKJEFT01
//STEPLIB DD DSN=DCF140.DCFLOAD,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
 SCRIPT 'WEC.DOC.SCRIPT(DEMO)'                -
     PROFILE('DCF140.EDFMAC1(EDFPRF40)')      -
     LIB('DCF140.EDFMAC1')                    -
     DEV(38PPN)                               -
     FPASSES(3)                               -
     INDEX                                    -
     MESSAGE(DELAY TRACE)                     -
     PAGE(FROM * TO *)                        -
     SYSVAR(D NB  H NO  X NO  Q 1  G INLINE)  -
     CONTINUE                                 -
     FONTLIB('DCF140.EDF38P1')                -
     FILE('WEC.SCRIPT.LIST38PP')
/*
```

Using BookManager/BUILD

While EVIEVM might have the famous **B2H** program, EVIEMVS has a working copy of BookManager. Running BookManager/BUILD is easy from ISPF -- simply run

```
BUILD
```

from the ISPF Command Shell (or TSO). Now, you will be presented with a series of dialogs, which you will need to fill in:

```
---------------------- BookManager BUILD/MVS Main Panel ---------------

Type the name of the data set from which to build a book:

ISPF library:
   PROJECT. . . wec
   GROUP. . . . doc
   TYPE . . . . script
   MEMBER . . . demo      (Blank or pattern for member selection list)

Other partitioned or sequential data set:
   Data set name. . .

Markup language (type x to select one):
   GML Starter Set          BookMaster X

BUILD options data set. . .

                                                          Press ENTER to co



COMMAND ===>
 F1=Help    F2=Split   F3=Exit    F6=Submit  F9=Swap   F12=Cancel
```

Figure 3. BookManager/BUILD/MVS Screen 1

Specify the input file, and you want to use BookMaster as the language (unless you really do want to use GML). Upon pressing Enter, you will be asked where you want to store the output dataset; the dataset name defaults to **userid.membername.BOOK**.

```
------------------------------- BUILD Options ------------------------

Markup language . . . . . : BookMaster
Output data set for book. . : 'WEC.DEMO.BOOK'

SCRIPT command name . . . . . . . . . . . . . . . . . . . . . . SCRI

TOPIC (type tag to use for lowest topic level). . . . . . . . . . :H4.

TRACE level (type 1-6). . . . . . . . . . . . . . . . . . . . . . 2

MSGNO tags create topics (type x to select) . . . . . . . . . . . .

Picture types (type 1-9 to specify search order):
  ADMGDF  .. 1    ADMIMG . . 2       CCITTG4. . 8       CGM . . . . 7
  EPS. .  .. 9    PSEG3820 . 3       PSEG38PP . 4       PSEG4250. . 5
  METAFILE . 6

 To see picture defaults, press F5.  To add picture data sets,
 position the cursor on the picture type field and press F4.

COMMAND ===>
 F1=Help      F2=Split     F3=Exit      F4=Add       F5=Defaults
 F9=Swap     F12=Cancel
```

Figure 4. BookManager/BUILD/MVS Screen 2

So, this is ultimately generating a batch job. This JCL is templated from **EOY.SEOYPROC(EOYBUILD)**, but I have already modified it to work. On the next dialog, press Enter:

```
--------------------------- SCRIPT Command Options --------------------

INDEX (type x to select). . . .
LIB data sets . . . . . .




MESSAGE (type x to select). . .   id     trace
NOPROF (type x to select) . . .
PROFILE data set. . .

SYSVAR. . . . . .

Additional SCRIPT options . . .



                                                        Press ENTER to co



COMMAND ===>
 F1=Help    F2=Split   F3=Exit    F6=Submit  F9=Swap   F12=Cancel
```

Figure 5. BookManager/BUILD/MVS Screen 3

Finally, enter the JCL parameters:

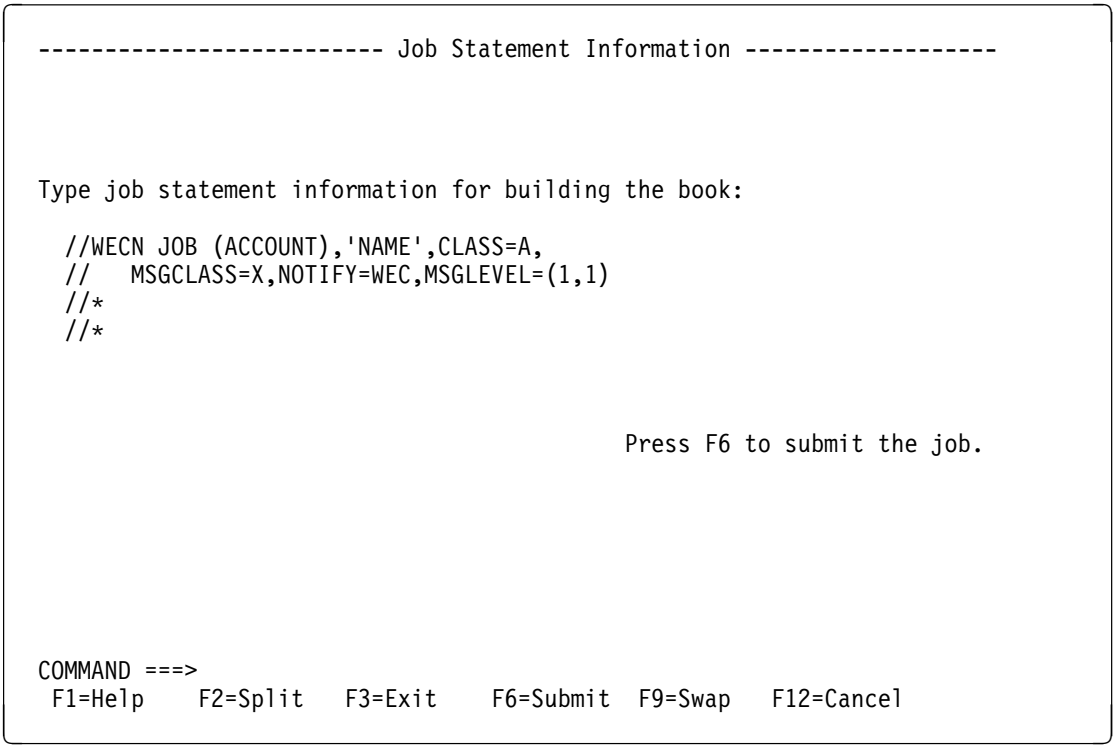

Figure 6. BookManager/BUILD/MVS Screen 4

When you view the job on SDSF, you will see that, outside of being slo it should run with no errors (unless you have formatting errors).

Using BookManager/READ

The counterpart of BookManager/BUILD is **BookManager/READ.** This is invoked from TSO by typing **BOOKMGR**:

```
  Books  View  Options  Help
--------------------------------------------------------------------------


                              Bookshelf List
                                                              Shelves 1 to
   Shelf Name  Description
__ BMBUILD     BookManager Build/MVS 1.3
__ BOOKMGRR    BookManager Read/MVS 1.3 Books.
__ BOOKMSTR    BookMaster 1.4
__ C370        C/370, C for VM/ESA, XL C/C++
__ OGL370      OGL/370 1.1
__ OVCAL       OfficeVision/VM ESA Calendar V2R1
__ OVVM        OfficeVision/VM R4.0
__ SCRIPT      SCRIPT/VS 1.4







Command ===> _____________________________________________ SCROLL ===
 F1=Help      F2=Split     F3=Exit      F5=Refresh   F6=Describe  F7=Bkw
 F8=Fwd       F9=Swap     F10=Actions  F11=Retrieve F12=Cancel
```

Figure 7. BookManager/READ/MVS Main Menu

Note that this bookshelf list can be accessed by going to *Books* then *Open a bookshelf* -- enter **WEC.EOXMBMGR.BKLSHELF** fo the dataset name.

To open the book you just produced, enter its dataset name on *Books* > *Open a Book*:

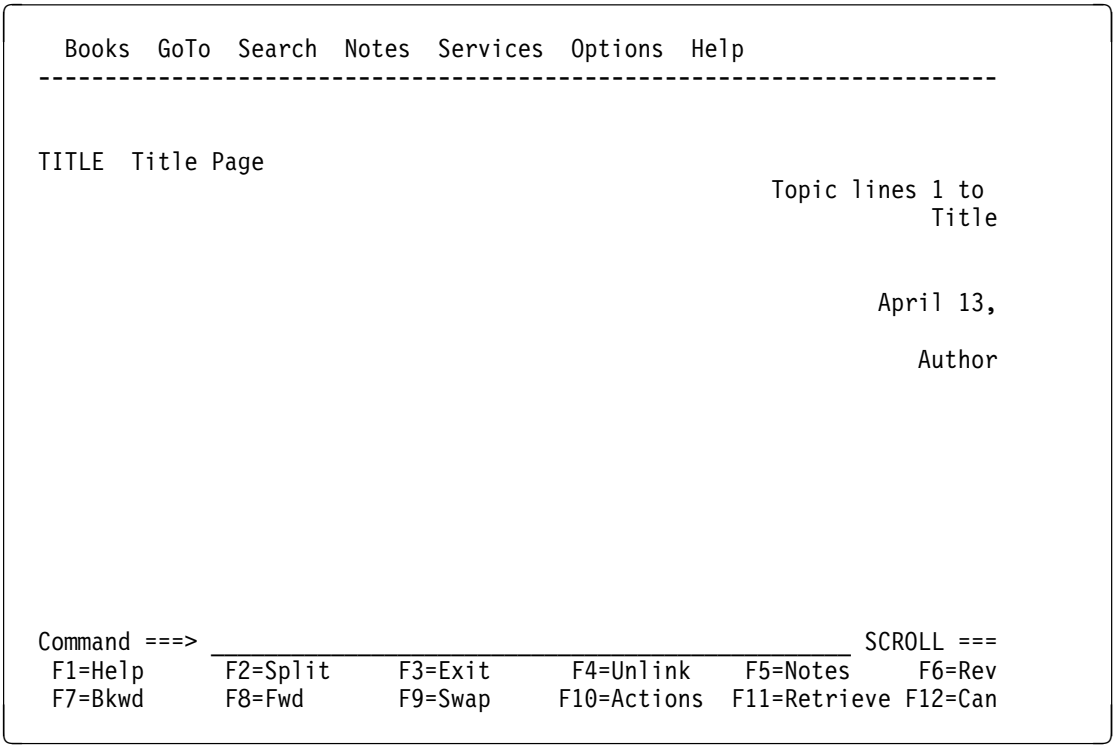

Figure 8. BookManager/READ/MVS Book Display

Easy as that!