SYSGEN/UNIX 7000

1060

Bell Laboratories

subject: SYSGEN Shell Procedure
Creating a New File System

date: July 25, 1974

from: I. A. Winheim

PROGRAMMING NOTES
CREATING A NEW FILE SYSTEM

I. Introduction

SYSGEN, a shell procedure to generate new systems, and to recompile portions of the generated systems, has been added to UNIX Release 2. Its full pathname is /usr/sys/conf/sysgen. This is the same shell file which generated the Unix Support Group (USG) "Frozen" system, and except for the fact that the instructions used to create future file systems are executed by the Frozen and not the running system, the same shell procedure is used to generate user systems.

II. Format of SYSGEN

There are three parameters to SYSGEN which make the resulting file system compatible with the machine on which it will be installed. The first parameter is the model (either 40 or 45). The second pertains to the floating point option on the 11/45. If the machine is an 11/40 the second parameter must be NOFPP. The same is true for 11/45 machines without the floating point option. For those machines with floating point an FPP is entered as the second parameter. The third parameter is the name of a file containing the list of devices in the configuration. Creating this file is discussed in the section entitled "Using SYSGEN".

III. Program Operation

The program MKCONF is executed using the third input parameter as a file containing the names of devices in the configuration. This makes it possible to select drivers in UNIX to be compatible with these devices. The result of the execution is two files, l.s and c.c (which are similar to low.s and conf.c respectively). The file l.s which contains low core input/output interrupt vectors is assembled and the result is moved to l.o. The file c.c, is the configuration table containing entries for I/O on block and character devices. This is compiled with the -c option to produce a c.o.

Now that the proper devices have been defined a test is made to determine whether the system is to reside on a 40 or 45 machine (the first input parameter). The program mch.s is edited to set a 40/45 flag correctly. The edited line should read

```
.45 = 0     /0 = 11/40     1 = 11/45
```

for an 11/40 or

```
.45 = 1     /0 = 11/40     1 = 11/45
```

for an 11/45.

The floating point option (second input parameter) is then tested. Once again mch.s is edited. This time to set the floating point flag to indicate the proper mode. The edited line in mch.s should read

```
.fpp = 0     /0 = nofpp     1 = fpp
```

for no floating point or

```
.fpp = 1     /0 = nofpp     1 = fpp
```

for the existence of the floating point option. The program mch.s is then assembled and the result is moved to mch.o. The object programs l.o, mch.o, c.o, /usr/sys/lib1, and /usr/sys/lib2 are loaded with the instruction

```
ld -x l.o mch.o c.o /usr/sys/lib?
```

The -x option causes only external symbols to be placed in the symbol table thus saving space. In addition the references to the archive files /usr/sys/lib? causes all appropriate system routines to be loaded through resolution of external symbols in l.o and conf.o. The initial preparations are completed and SYSGEN will now execute this program.

MKFS constructs the file system on an RK pack according to the directions found in the file called proto. The description of the contents of the proto file is contained in the UNIX User's Manual but summarized below for completeness. Proto contains tokens separated by spaces, tabs, or newlines. The first token is the name of the file to be copied onto block 0 as the boot program. The second token refers to the size (in blocks) of the raw file system to be created. The next token is the i-list size in blocks. The next line comprises the specification for the root directory. The mode, user-id, group-id, and the initial contents of the file tokens are included. The mode of all file entries indicates whether the files are regular, block special, character special, or directory special by the first character being -,b,c,d respectively. Most files in the proto used for the system generation are either regular or directory special. The second character of the mode token is used to indicate set-user-id mode (u) or not (-). The third character is either g or - for set-group-id mode or not. The proto used for system generations has no files with a set-group-id mode. The rest of the mode token consists of three numbers indicating read, write, execute permissions for owner, group and all others respectively. The two decimal number tokens following the mode indicate the user and group id's of the owner of the file. If the file is a regular file the next token is the pathname of the file that is being copied from the Frozen system into the new system. The contents of a directory are ended with the token $. The $ must exist and applies only to the preceding directory. This notation allows for "nesting" of directories within other directories. For more information on MKFS see the User Manual Section VII. A listing of the proto file used in the system generation is delivered with all new systems so that users can determine what was included in their system and where it resides.

Upon completion of MKFS the file system with a copy of the Frozen source and object code exists. SYSGEN executes the run file, FLRC, on systems that have no floating point . This sequence recompiles pass I of the C compiler (c0?.c where c0?.c matches c00.c, c01.c, c02.c, c03.c, c04.c, and c0h.c) and the debugger (db?.s) so they do not use floating point. It also installs the floating point interpreter (fp?.s) in the standard library (/lib/lib.a), and it reloads basic (bas?.s) and fortran (f?o.a) to have their standard floating point interpreted. In order to check the consistency of the new system, FLRC is executed on the file system that has been just created by using the commands in that file system. The sysgen procedure echos "system completed" on the TTY, when this process is done.

The line printer is used to print the following documentation:

configuration (device names)
sysgen shell procedure
configuration table (conf.c)
low core image (low.s)
a copy of the Proto file used for this system
a name list for the UNIX system built

The whole procedure takes about 15 minutes.

IV. Using SYSGEN

The purpose of the SYSGEN shell procedure is to control and simplify the generation of a base system which contains all the supported commands and features in Release 2 UNIX. The first step in generating user systems was to establish a Frozen system from which all subsequent systems were built. Improvements to the Frozen system are delivered on modification tapes and installed at the site of the machine.

To generate a new Frozen system a configuration file is used which contains names of the devices for the machine configuration. In general the device names used for the configuration file (with MKCONF program) are selected from the following list:

Name	Description
rk	DEC single platter disk cartridge drive
rp	2314 double density disk pack drive
rf	fixed head disk
tm	magnetic tape
tc	DEC tape
dhdm	asynchronous serial line multiplexer/modem control
dh	asynchronous serial line multiplexer
dc	asynchronous serial line interface
lp	high speed line printer
pc	high speed paper tape reader/punch
dp	synchronous interface
dn	automatic calling unit interface
kl	standard console interface

If you have more than one controller of a particular type, then the number precedes the device name. For example:

4dc

creates four vector slots in l.s for line multiplexers.

One kl(standard console interface), one clock interface, and device memory are automatically provided. For each block device specified, an entry is also created to allow physical I/O on that device.

The order in which devices appear in the file is important. Block devices always precede the character devices. If there are floating vector devices (one of the following):

dc11
kl11
dl11
dp11
dm11
dn11
dr11
pa511 Reader
pa611 Punch
dt11
dx11
kj11
dh11
gt40
lps11
vt20

in your machine configuration check /usr/sys/conf/low.s to make sure the hardware and the software vector positions correspond. If they do not match, edit the low.s file in the /usr/sys/conf directory to match the hardware, reassemble low.s and reload UNIX. (See /usr/sys/ld). Other devices may exist on your machine that are not present in the MKCONF program. They can be added by hand to conf.c and low.s according to their descriptions. These can be found in the DEC Peripherals Handbook.

The next step is to edit SYSGEN if you do not want your generated system on /dev/rp8, or if the node on which the system is to be mounted is not "mnt". The lines to edit are:

```
        /etc/mkfs       /dev/rp8       proto
        check      /dev/rp8
        /etc/mount      /dev/rp8        ***node***
        /etc/umount      /dev/rp8
```

After this is done execute the SYSGEN shell as super-user.

The SYSGEN shell will produce a copy of UNIX Release 2. At this point any files that are not to be distributed as UNIX Release 2, but are needed in the Frozen system can be added to the system but should not be placed in the proto file. The only change necessary in the SYSGEN proto file to use it as a Frozen system to build user systems, is to add the mounting node (in this case /mnt) in front of the pathnames of all the files. By doing this all files for new systems are taken from the Frozen system. Similarly the

SYSGEN file was changed to use the Frozen system as appropriate. Appendix A is a listing of the edited version of SYSGEN, with the changes to use the Frozen system to build user systems.

Run SYSGEN as the super-user so that the FLRC procedure can edit protected files. The frozen system must also be mounted under the proper node. It should also be noted that if your Frozen system does not have the floating point option, future file systems will not have the option. Modules affected by the floating point option can be obtained from the USG group. These modules should allow the user to execute SYSGEN on a no floating point system and produce a new system with floating point.

These notes describe the shell file that was designed and built to aid in the construction of user systems. This procedure can also be used by projects who in turn create a "Frozen" system and generate user systems to be sent to their customers.

I.A. Winheim

MH-8234-IAW-dh

I. A. Winheim

Copy to
G. L. Baldwin
M. M. Rochkind
G. W. R. Luderer
H. S. London
J. J. Ludwig
J. F. Maranzano
K. Brandt
J. Cloutier
W. Coston
R. Crane
J. Defelice
P. Kennedy
G. Foley
A. G. Fraser
R. Haight
P. Hamilton
J. Jarvis
H. Lycklama
F. Menzel
J. Molinelli
T. O'Connell
R. Pawnyk
R. Perdue

H. Pierson
D. M. Ritchie
D. Sandel
J. I. Sheetz
D. Sidor
C. A. Strohecker
T. Tabloski
B. A. Tague ← THIS COPY FOR
K. Thompson
R. White

```
cp /mnt/usr/sys/conf/mch.s mch.s
/mnt/usr/sys/conf/mkconf < $3
/mnt/bin/as l.s
/mnt/bin/mv a.out l.o
/mnt/bin/cc -c c.c
if $1 = 40 goto 40
if $1 != 45 goto error0
/mnt/bin/ed - mch.s
/45/c
.45 = 1     / 0 = 11/40 1 = 11/45
.
w
q
echo 11/45 >> $3
goto fpptst
: error0
echo 40/45 argument bad - sysgen aborted.
exit
: 40
/mnt/bin/ed - mch.s
/45/c
.45 = 0     / 0 = 11/40 1 = 11/45
.
w
q
echo 11/40 >> $3
: fpptst
if $2 = fpp goto fpp
if $2 != nofpp goto error1
/mnt/bin/ed - mch.s
/fpp/c
.fpp = 0     /0 = nofpp 1 = fpp
.
w
q
echo no fpp >> $3
goto cont
: error1
echo fpp argument bad - sysgen aborted.
exit
: fpp
/mnt/bin/ed - mch.s
/fpp/c
.fpp = 1     /0 = nofpp 1 = fpp
.
w
q
echo fpp >> $3
: cont
pr -h "configuration" $3 |lpr
ed - $3
$d
$d
w
q
pr -h "sysgenx shell procedure" /mnt/usr/sys/conf/sysgenx | lpr
pr -h "protox used for sysgen" /mnt/usr/sys/conf/protox | lpr
pr -h "configuration table" c.c | lpr
pr -h "low core image " l.s | lpr
/mnt/bin/as mch.s
/mnt/bin/mv a.out mch.o
/mnt/bin/ld -x l.o mch.o c.o /mnt/usr/sys/lib?
nm |sort -n |pr -h "name list" -2 |lpr
echo make file system started
/mnt/etc/mkfs  /dev/rk0 /mnt/usr/sys/conf/protox
check /dev/rk0
rm a.out c.c c.o l.o l.s mch.o
if $2 = fpp goto tape
/etc/mount /dev/rk0 /newsys
chdir /newsys/usr/c
/newsys/bin/ed - c0t.s
/fpp/g/1/s//0/
w
q
/newsys/bin/as c0t.s
/newsys/bin/mv a.out c0t.o
/newsys/bin/cc -n c0?.o
strip a.out
cp a.out /newsys/lib/c0
rm a.out c0t.o
chdir /newsys/usr/source/s1
/newsys/bin/ed - db1.s
/fpp/g/1/s//0/
w
q
/newsys/bin/as db?.s
strip a.out
cp a.out /newsys/bin/db
rm a.out
chdir /newsys/usr/source/s3
/newsys/bin/as fp?.s
/newsys/bin/mv a.out fp.o
ar r /newsys/lib/liba.a fp.o
rm fp.o
chdir /newsys/usr/fort
/newsys/bin/ld -u pass1 -u pass2 -u pass3 -u pass4\
f1/f1o.a f2/f2o.a f3/f3o.a f4/f4o.a fx/fxo.a -l
strip a.out
cp a.out fc1
rm a.out
chdir /newsys/usr/source/s1
/newsys/bin/as bas?.s
/newsys/bin/ld a.out -l
strip a.out
cp a.out /newsys/bin/bas
rm a.out
/etc/umount /dev/rk0
check rk0
: tape
cat /mnt/usr/mdec/mt0 /dev/rk0 > /dev/mt0
/etc/umount /dev/rp3
echo sysgen done thank God
```