OSVS1 Sysgen

I would first like to thank three individuals for their hard work in getting a version of OSVS1 available. Kevin Leonard, Mark Waterbury, and Moshix. Thank you very much for your contributions to the hobbyist community. Lastly IBM for creating the OS. This document assumes familiarity with Hercules, telnet, 3270 emulation, and a general understanding of the process. Going through the MVS generation process helped me with this project. My thanks to Jay Moseley for his excellent efforts in that regard. Also, a thank you to both Moshix (Moshix Mainframe channel) and Matthew (Mainframes & more with Matthew channel) for their instructional videos. I highly recommend them on YouTube if you are interested in Mainframes and a different kind of retro-computing. This guide is based heavily on the README.md in the Moshix repository.

Now, down to business.

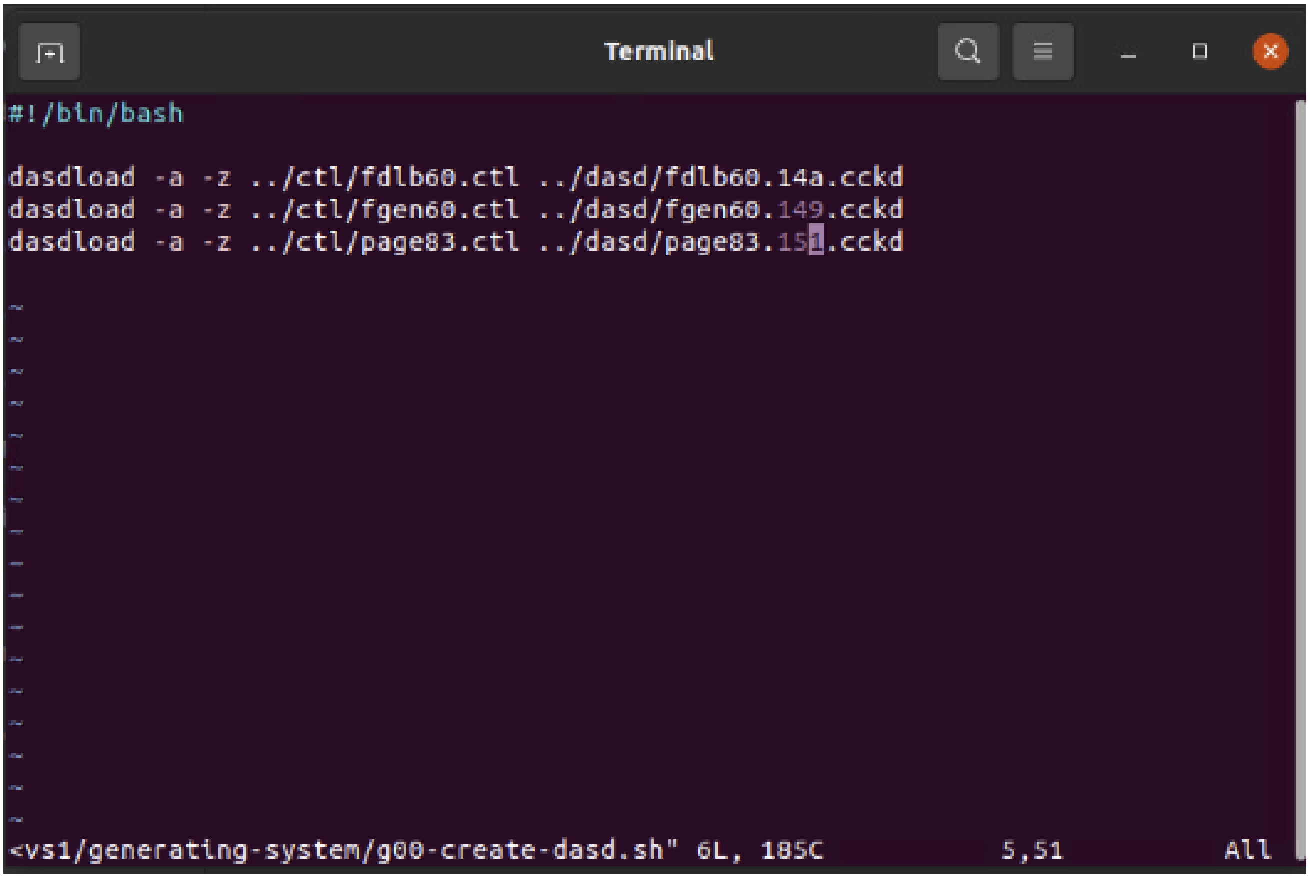
Prerequisites:

1. A clone or zip file of the Moshix OSVS1 repository.
   https://github.com/moshix/OSVS1
2. A working version of Hercules. (I evaluated this on SDL Hercules Hyperion 4.3)
3. A telnet client (I used the telnet baked into Ubuntu 20.04)
4. A 3270 terminal emulator. (I use c3270 in this example)

I completed this entire process on Ubuntu 20.04 Desktop.

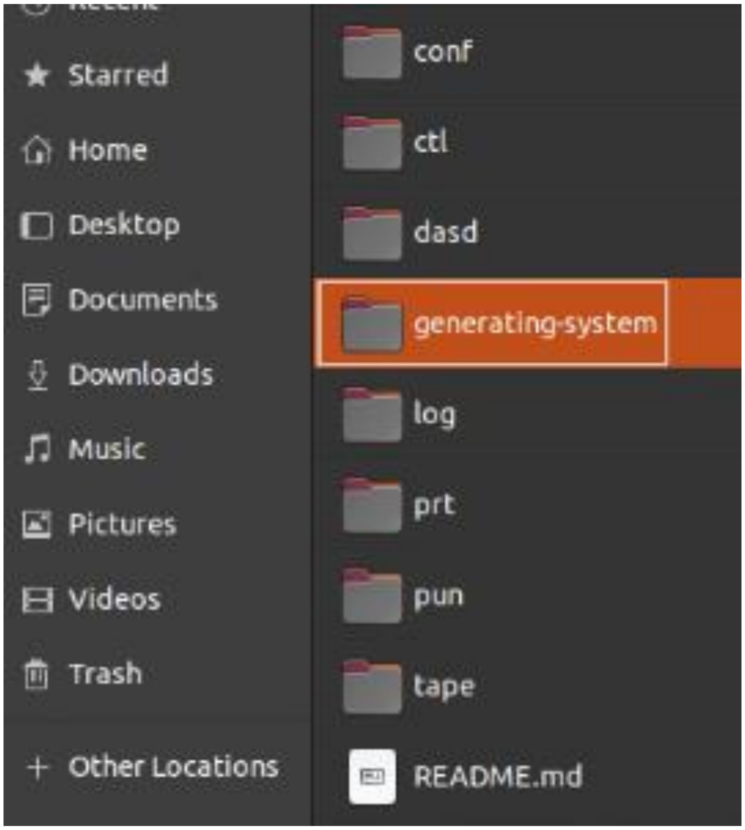
If you use a different OS YMMV.

I also created a Linux script to create the DASD needed for the process. See below:

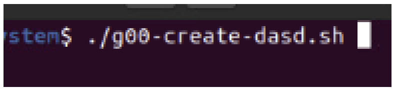


```
#!/bin/bash

dasdload -a -z ../ctl/fdlb60.ctl ../dasd/fdlb60.14a.cckd
dasdload -a -z ../ctl/fgen60.ctl ../dasd/fgen60.149.cckd
dasdload -a -z ../ctl/page83.ctl ../dasd/page83.151.cckd


~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
<vs1/generating-system/g00-create-dasd.sh" 6L, 185C          5,51          All
```

1. Unzip or clone the repository to a folder of your choosing. Once you finish the extractions you will see a setup like this: (NOTE: Be sure to place the tape files in the tape folder)

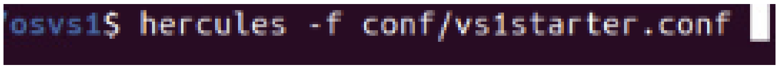


2. The next step is to create the DASD we will need to complete the process. Open a terminal in the generating-system folder and run the shell script to create the DASD.



```
stem$ ./g00-create-dasd.sh
```

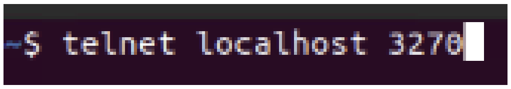
3. Move up a directory with the command cd ..
4. We will then start Hercules with the command hercules -f conf/vs1starter.conf



```
osvs1$ hercules -f conf/vs1starter.conf
```

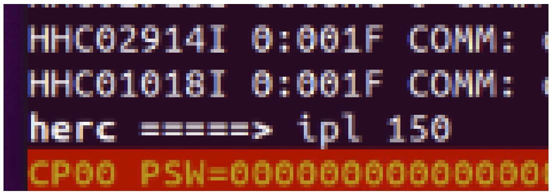
(NOTE: Be sure that all the other DASDs are commented out except for dliba1.150.cckd in the config file.)

5. Open a second terminal window and enter telnet localhost 3270 to connect a console to Hercules.



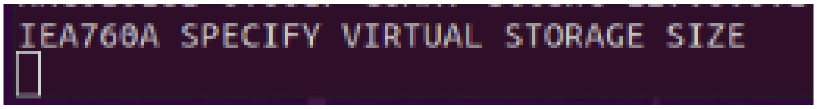
```
~$ telnet localhost 3270
```

6. To initiate the starter system, we need to run ipl 150 from the Hercules console.



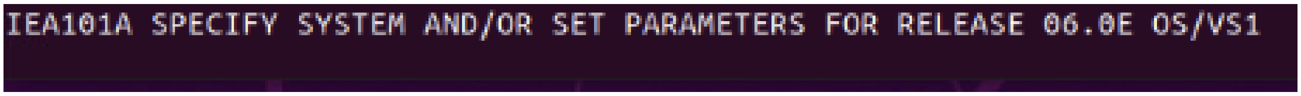
```
HHC02914I 0:001F COMM:
HHC01018I 0:001F COMM:
herc =====> ipl 150
CP00 PSW=0000000000000000
```

7. When you see this pop-up press Enter:



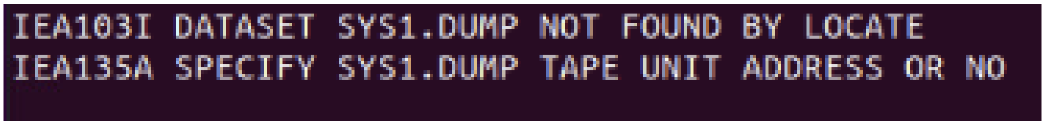
```
IEA760A SPECIFY VIRTUAL STORAGE SIZE
```

8. Press Enter again for this message:



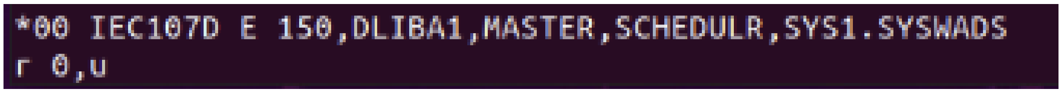
```
IEA101A SPECIFY SYSTEM AND/OR SET PARAMETERS FOR RELEASE 06.0E OS/VS1
```

9. Press Enter one last time for this message:



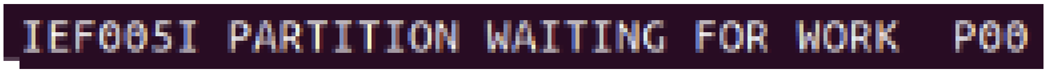
```
IEA103I DATASET SYS1.DUMP NOT FOUND BY LOCATE
IEA135A SPECIFY SYS1.DUMP TAPE UNIT ADDRESS OR NO
```

10. The next message that stops us is one that requires a reply. Reply with r 0,u



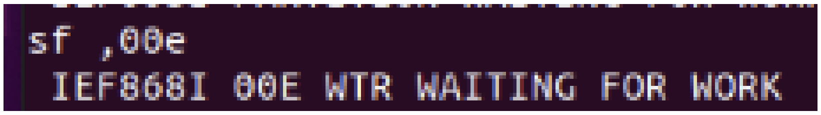
```
*00 IEC107D E 150,DLIBA1,MASTER,SCHEDULR,SYS1.SYSWADS
r 0,u
```

11. You should now have a message saying PARTITION WAITING FOR WORK P00.



```
IEF005I PARTITION WAITING FOR WORK  P00
```

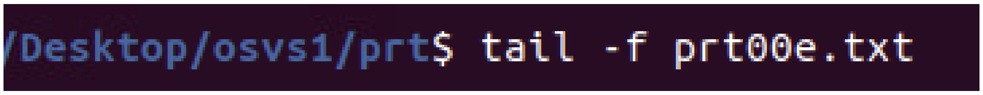
12. Start the printer with the command sf ,00e



```
sf ,00e
 IEF868I 00E WTR WAITING FOR WORK
```

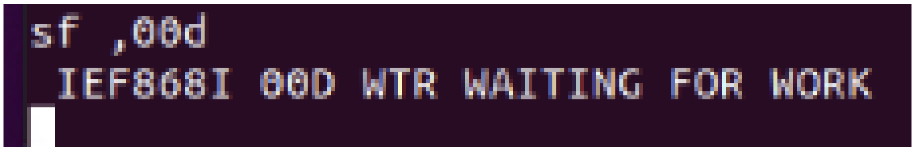
13. Open a new terminal window and go to the prt folder. Then enter the command tail -f prt00e.txt

    This will give you a printout for all the jobs so that you can scan them for completion codes.



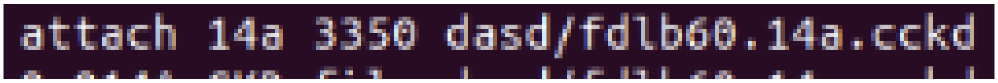
```
/Desktop/osvs1/prt$ tail -f prt00e.txt
```

14. Start the card punch with sf ,00d



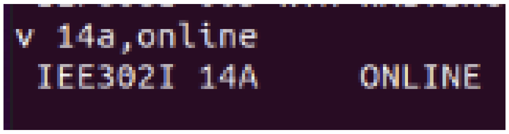
```
sf ,00d
 IEF868I 00D WTR WAITING FOR WORK
```

15. Attach the fdlb60.14a.cckd drive to Hercules with the command attach 14a 3350 dasd/fdlb60.14a.cckd



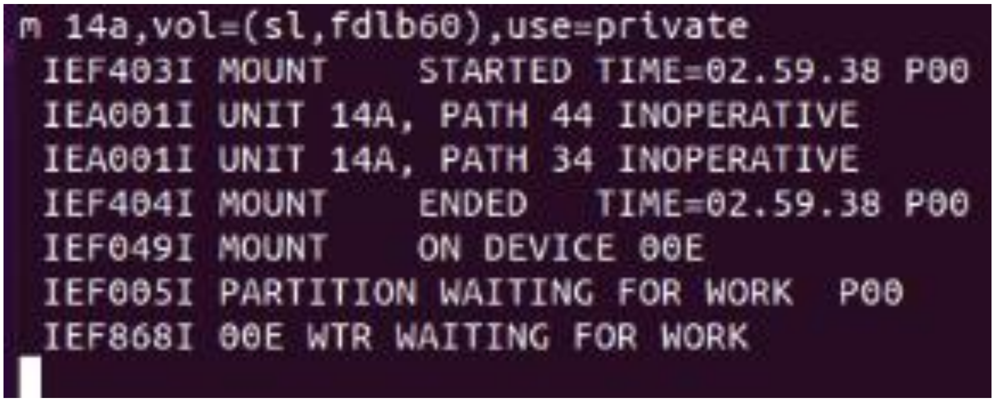
```
attach 14a 3350 dasd/fdlb60.14a.cckd
```

16. In the telnet terminal vary the drive online with the command v 14a,online



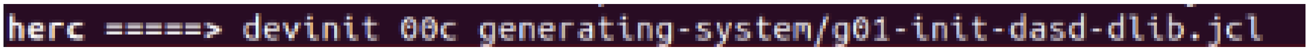
```
v 14a,online
 IEE302I 14A      ONLINE
```

17. Mount the drive with the command m 14a,vol=(sl,fdlb60),use=private



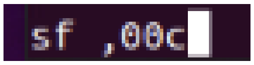
```
m 14a,vol=(sl,fdlb60),use=private
 IEF403I MOUNT    STARTED TIME=02.59.38 P00
 IEA001I UNIT 14A, PATH 44 INOPERATIVE
 IEA001I UNIT 14A, PATH 34 INOPERATIVE
 IEF404I MOUNT    ENDED   TIME=02.59.38 P00
 IEF049I MOUNT    ON DEVICE 00E
 IEF005I PARTITION WAITING FOR WORK  P00
 IEF868I 00E WTR WAITING FOR WORK
```

18. In the Hercules console initialize the card reader with devinit 00c generating-system/g01-init-dasd-dlib.jcl



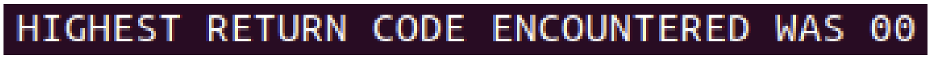
```
herc =====> devinit 00c generating-system/g01-init-dasd-dlib.jcl
```

19. In the telnet terminal run sf ,00c to start the job (you must only do this the first time to have the reader function automatically thereafter)



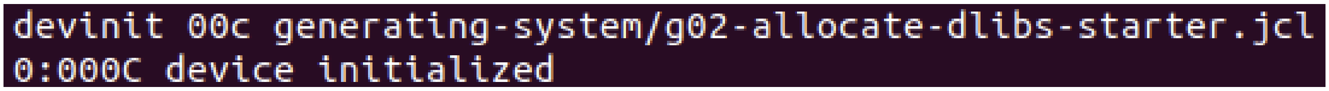
```
sf ,00c
```

20. In the printer window you should see that the highest return code encountered was 00.



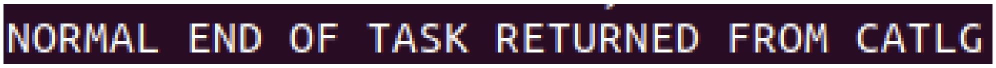
```
HIGHEST RETURN CODE ENCOUNTERED WAS 00
```

21. Next, we allocate the DLIB data sets by issuing the following command in the Hercules console devinit 00c generating-system/g02-allocate-dlibs-starter.jcl



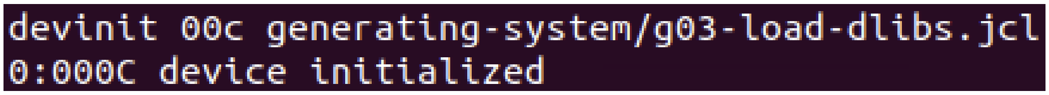
```
devinit 00c generating-system/g02-allocate-dlibs-starter.jcl
0:000C device initialized
```

22. In the printer window the job should show NORMAL END OF TASK for each of the datasets. (NOTE: You may have to scroll up in the terminal window to see these results)



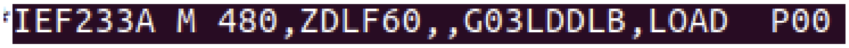
```
NORMAL END OF TASK RETURNED FROM CATLG
```

23. The next job loads the DLIBs from a tape. Start the job by issuing devinit 00c generating-system/g03-load-dlibs.jcl



```
devinit 00c generating-system/g03-load-dlibs.jcl
0:000C device initialized
```

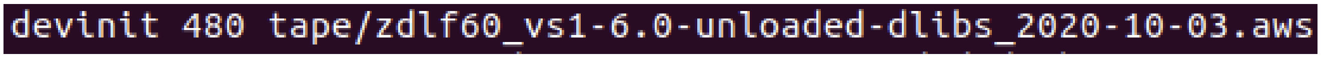
24. On the OSVS1 console you will see the following user-surly message:



```
IEF233A M 480,ZDLF60,,G03LDDLB,LOAD  P00
```

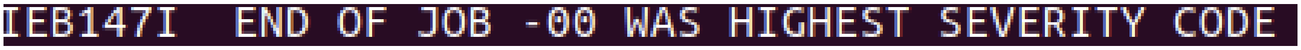
Which informs us that we need to mount a tape on device 480 with the label of ZDLF60.

To mount the tape, we use the devinit command again in the Hercules console: devinit 480 tape/zdlf60_vs1-6.0-unloaded-dlibs_2020-10-03.aws



```
devinit 480 tape/zdlf60_vs1-6.0-unloaded-dlibs_2020-10-03.aws
```

25. When the job finishes, we should see that 00 WAS THE HIGHEST SEVERITY CODE in the printer window



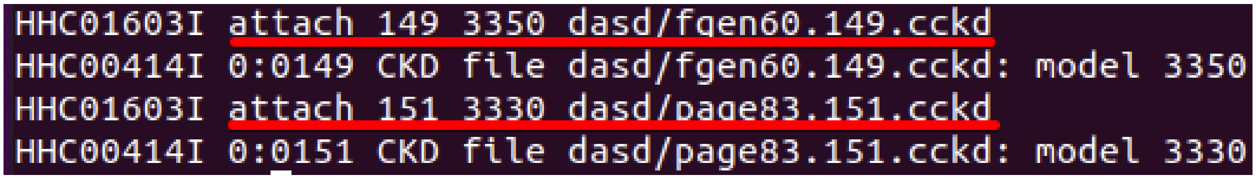
```
IEB147I  END OF JOB -00 WAS HIGHEST SEVERITY CODE
```

26. We now turn our attention to the other DASDs created earlier.

    Again, we start with attaching them to the mainframe in the Hercules console with the commands:

    attach 149 3350 dasd/fgen60.149.cckd

    attach 151 3330 dasd/page83.151.cckd

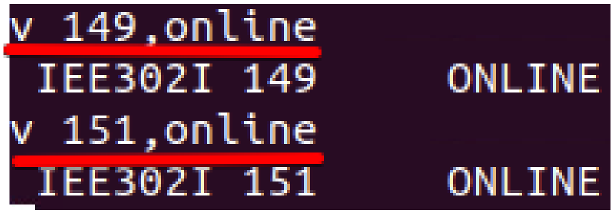


```
HHC01603I attach 149 3350 dasd/fgen60.149.cckd
HHC00414I 0:0149 CKD file dasd/fgen60.149.cckd: model 3350
HHC01603I attach 151 3330 dasd/page83.151.cckd
HHC00414I 0:0151 CKD file dasd/page83.151.cckd: model 3330
```

27. Now we vary both online in the starter console with:

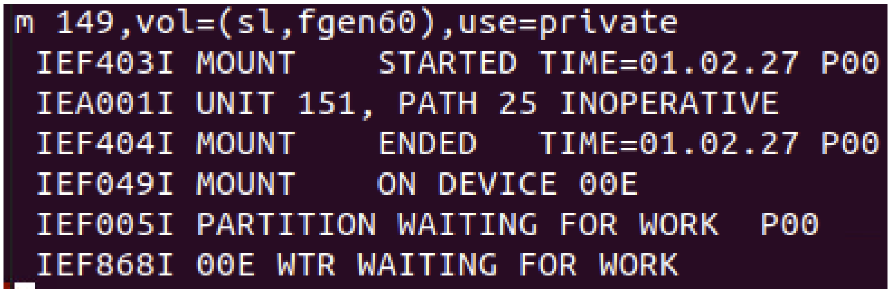
    v 149,online

    v 151,online



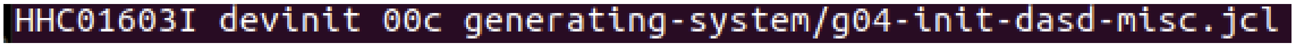
```
v 149,online
 IEE302I 149      ONLINE
v 151,online
 IEE302I 151      ONLINE
```

28. Now, we mount only the residence dasd fgen60 by issuing this command: m 149,vol=(sl,fgen60),use=private



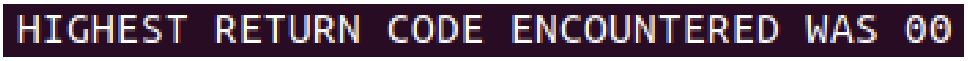
```
m 149,vol=(sl,fgen60),use=private
 IEF403I MOUNT     STARTED TIME=01.02.27 P00
 IEA001I UNIT 151, PATH 25 INOPERATIVE
 IEF404I MOUNT     ENDED   TIME=01.02.27 P00
 IEF049I MOUNT     ON DEVICE 00E
 IEF005I PARTITION WAITING FOR WORK   P00
 IEF868I 00E WTR WAITING FOR WORK
```

29. Time for the next card set. We initialize the two dasd via the Hercules console with: devinit 00c generating-system/g04-init-dasd-misc.jcl



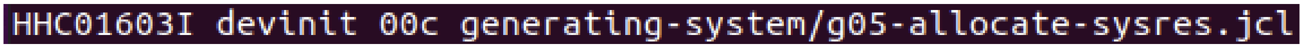
```
HHC01603I devinit 00c generating-system/g04-init-dasd-misc.jcl
```

30. This job should have a return code of 00 when finished.



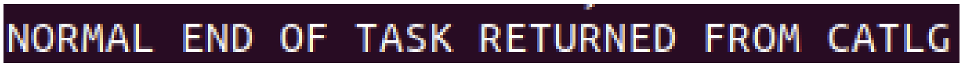
```
HIGHEST RETURN CODE ENCOUNTERED WAS 00
```

31. Next, we allocate the system datasets with: devinit 00c generating-system/g05-allocate-sysres.jcl



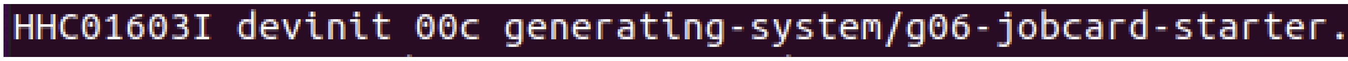
```
HHC01603I devinit 00c generating-system/g05-allocate-sysres.jcl
```

32. All the catlg steps should return NORMAL END OF TASK from this job (NOTE: You may have to scroll up in the terminal window to see these results)



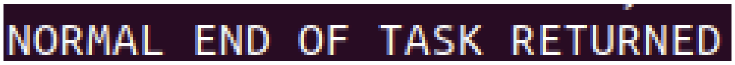
```
NORMAL  END  OF  TASK  RETURNED  FROM  CATLG
```

33. The next job modifies the JOBCARD sysgen macro. Start it by executing the following in the Hercules console: devinit 00c generating-system/g06-jobcard-starter.jcl



```
HHC01603I devinit 00c generating-system/g06-jobcard-starter.
```

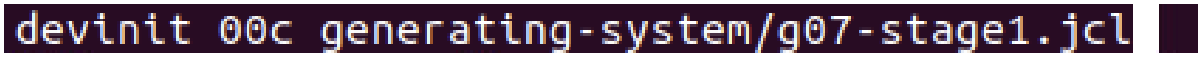
34. You get two messages from the job in the printer window showing that there was a normal end of task for both the scratch and rename tasks.



```
NORMAL  END  OF  TASK  RETURNED
```

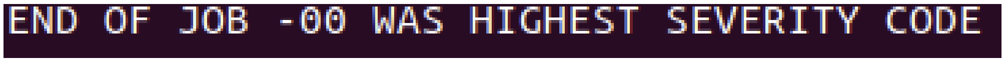
35. In the next step we enter stage 1 of the actual sysgen.

    By editing the jcl for this job you can modify the configuration of the final system. When you finish reviewing/editing submit the job in the Hercules console with: devinit 00c generating-system/g07-stage1.jcl



```
devinit 00c generating-system/g07-stage1.jcl
```

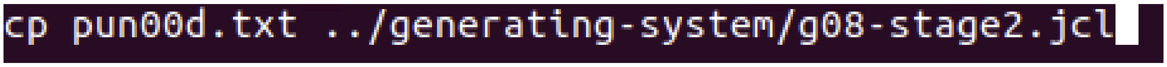
36. The job should say that the HIGHEST SEVERITY WAS 0



```
END  OF  JOB  -00  WAS  HIGHEST  SEVERITY  CODE
```

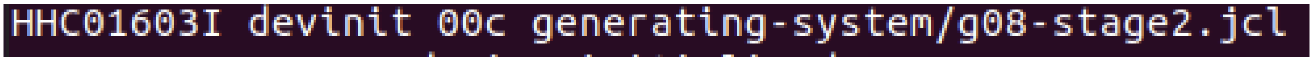
37. The previous job punches out the jobs for the next steps.

    Open a terminal window to the pun folder and issue the following command to copy the punched output to a jcl file in the generating-system folder.: cp pun00d.txt ../generating-system/g08-stage2.jcl



```
cp pun00d.txt ../generating-system/g08-stage2.jcl
```

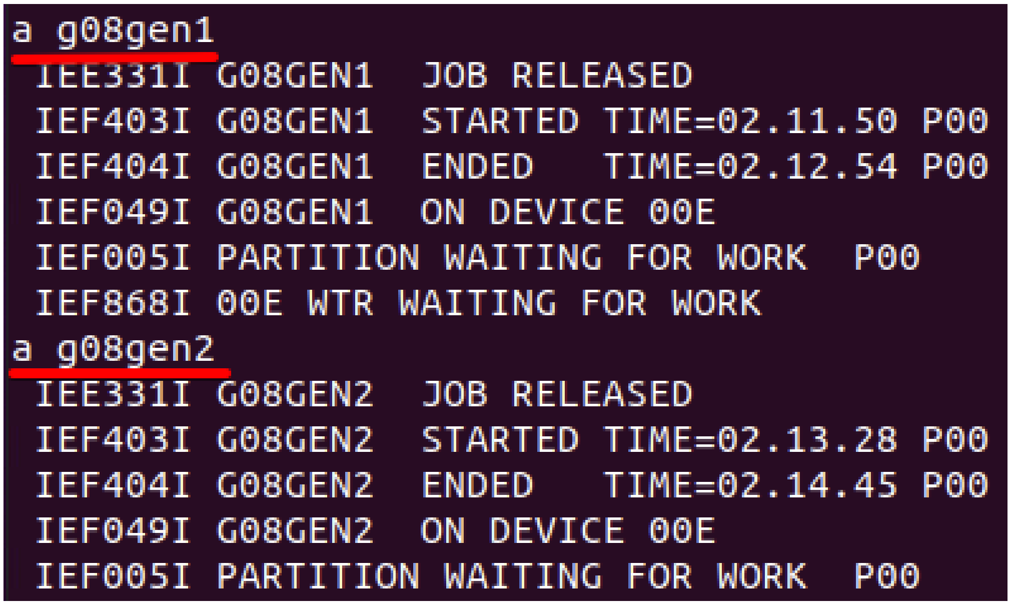
38. After copying the jcl submit the job in the Hercules console with: devinit 00c generating-system/g08-stage2.jcl



```
HHC01603I devinit 00c generating-system/g08-stage2.jcl
```

39. The previous step places fifteen held jobs in the spool. Release them one at a time in the starter console with:

(NOTE: remember to let each job finish before releasing the next)



```
a g08gen1
 IEE331I G08GEN1  JOB RELEASED
 IEF403I G08GEN1  STARTED TIME=02.11.50 P00
 IEF404I G08GEN1  ENDED   TIME=02.12.54 P00
 IEF049I G08GEN1  ON DEVICE 00E
 IEF005I PARTITION WAITING FOR WORK  P00
 IEF868I 00E WTR WAITING FOR WORK
a g08gen2
 IEE331I G08GEN2  JOB RELEASED
 IEF403I G08GEN2  STARTED TIME=02.13.28 P00
 IEF404I G08GEN2  ENDED   TIME=02.14.45 P00
 IEF049I G08GEN2  ON DEVICE 00E
 IEF005I PARTITION WAITING FOR WORK  P00
```

a g08gen1

a g08gen2

a g08gen3

a g08gen4

a g08gen5

a g08gen6

a g08gen7

a g08gen8

a g08gen9

a g08gen10

a g08gen11

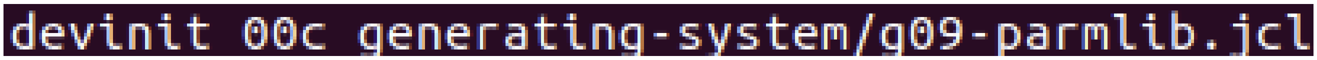
a g08gen12

a g08gen13

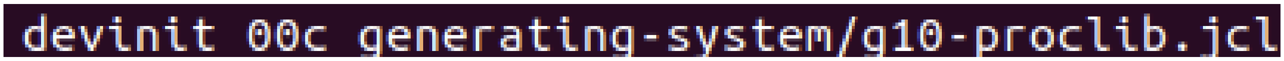
a g08gen14

a g08gen15

40. Each job should say that the HIGHEST SEVERITY WAS 0, 00 WAS THE HIGHEST SEVERITY CODE,
    COND CODE 0000, HIGHEST CONDITION CODE WAS 00000000, or NORMAL END OF TASK
    (NOTE: Jobs 9, 11, 12, and 15 do not seem to have any discernable condition codes. Nothing blew up so it I assume that they worked properly 😉)
41. The next step is another point in the process where you can make customizations.
    Review/Edit this file to set IPL parameters.
    Submit the job in the Hercules console when you are ready with: devinit 00c generating-system/g09-parmlib.jcl

    ```
    devinit 00c generating-system/g09-parmlib.jcl
    ```
42. The next step is our last submitted job. It adds procedures to SYS1.PROCLIB.
    Submit it with: devinit 00c generating-system/g10-proclib.jcl

    ```
    devinit 00c generating-system/g10-proclib.jcl
    ```
43. Issue the following commands in the system console to shut down the starter system:
    p 00c
    p 00d
    p 00e
    p p0
    z eod
44. In the Hercules console enter the quit command.
45. Reconfigure the mainframe by editing the Hercules config file.
    comment out the 150 dasd
    uncomment the three other dasd on 149, 14a, and 151
    comment out the 3215 console
    uncomment the 3270 console and the hard copy
46. To IPL the new system start Hercules with hercules -f conf/vs1starter.conf
47. Connect your 3270 console of choice to port 3270 on localhost. (I used c3270)

48. Start the system: ipl 149

49. Press enter at the specify virtual storage prompt.

50. When asked to specify the system parameters enter: r 0,’auto=cold’

51. Press enter when asked to specify the dump options.

52. Your new OSVS1 mainframe is ready to go!!