

# Linux for S/390

## Installation - Part 1

Richard Lewis (rflewis@us.ibm.com)  
Chuck Morse (morsec@us.ibm.com)



## Disclaimer



The information contained in this document has not been submitted to any formal IBM test and is distributed on an "AS IS" basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

In this document, any references made to an IBM licensed program are not intended to state or imply that only IBM's licensed program may be used; any functionally equivalent program may be used instead.

Any performance data contained in this document was determined in a controlled environment and, therefore, the results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environments.

It is possible that this material may contain reference to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM products, programming or services in your country.

Permission is hereby granted to WAVV Inc. to publish an exact copy of this paper in the conference proceedings. IBM retains the title to the copyright in this paper as well as title to the copyright in all underlying works. IBM retains the right to make derivative works and to republish and distribute this paper to whomever it chooses in any way it chooses.



# Trademarks



The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

IBM*	MQSeries*	z/VM
IBM logo*	DB2 Universal	z/OS
S/390**	Database	zSeries
e-business logo*	Multiprise*	z/Architecture
VM/ESA*	ESCON*	FICON
DB2*	VSE/ESA	
CICS*	WebSphere	
	DB2 Connect	

\* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation  
Tivoli is a trademark of Tivoli Systems Inc.  
LINUX is a registered trademark of Linus Torvalds.  
Penguin (Tux) compliments of Larry Ewing.  
Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries  
UNIX is a registered trademark of The Open Group in the United States and other countries.  
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.  
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.



# Agenda



- Part 1
  - Introduction
  - Installing the Marist system (VM and LPAR)
- Part 2
  - Installing the SuSE beta (VM and LPAR)
  - Administration and Maintenance



# Introduction



## What Makes up a Linux System?



- Linux kernel
  - the operating system core
- File system
  - everything else
    - utilities
    - configuration files
    - applications
- Parameter file
- IPL records



# Linux for S/390



## What's Available?

- Today
  - S/390 source updates to Linux
  - Marist College joint-study "distribution"
  - SuSE Beta 2 distribution
  - Over 500 applications
- In the Future
  - SuSE GA distribution (4Q00)
  - TurboLinux beta and GA distributions (4Q00?)



# Linux for S/390



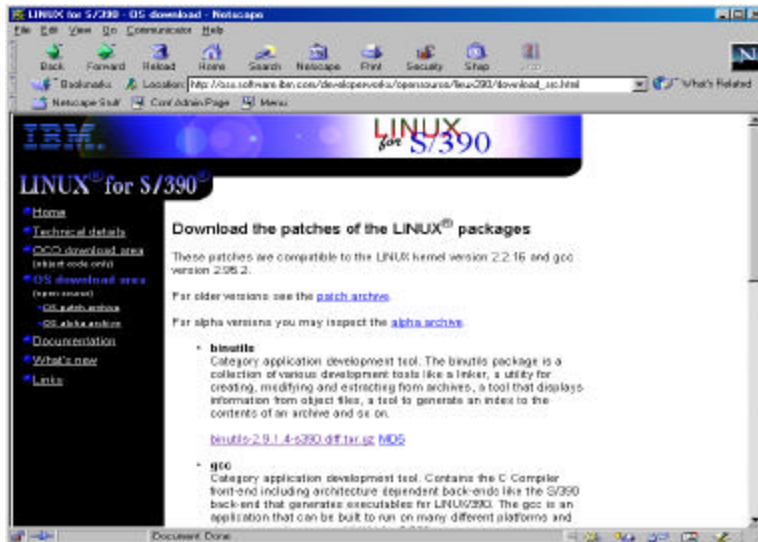
## Source Updates

- S/390 source patches to:
  - GCC Compiler
  - Kernel
  - Binutils
  - GNU Runtime Library
  - GNU Debugger
  - Strace

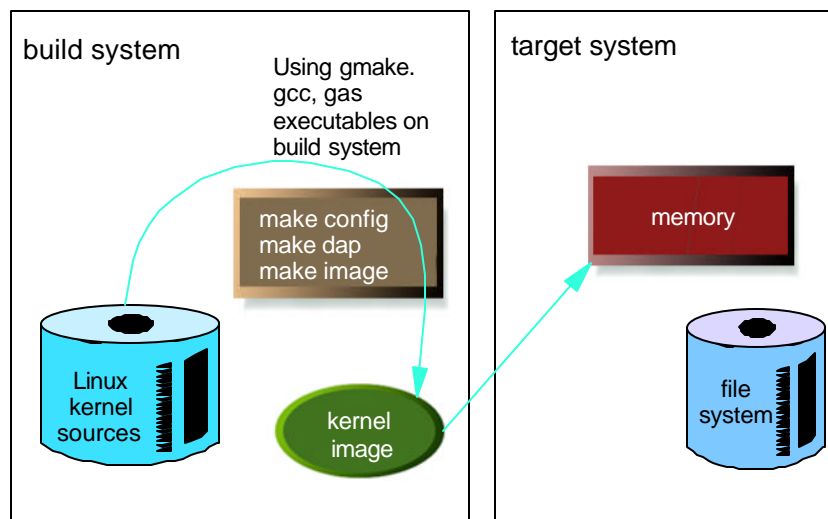
[http://oss.software.ibm.com/developerworks/opensource/linux390/download\\_src.html](http://oss.software.ibm.com/developerworks/opensource/linux390/download_src.html)



# IBM DeveloperWorks Site



# Cross Compile Build Process



# Linux for S/390 Environments



- S/390 or zSeries 900 single image
- S/390 or zSeries LPAR
- Virtual Image Facility
- VM/ESA or z/VM



# Linux for S/390

## Basic hardware Requirements



- Processors
  - zSeries 900
  - 9672 G2 - G6
  - Multiprise 2000
  - Multiprise 3000
  - Integrated Server
  - P/390 & R/390
- At least 64MB central storage



# Linux for S/390



## Basic hardware Requirements ...

- Network Connection
  - Network Adapters using LAN Channel Station (LCS) Protocol (token ring, ethernet, fast ethernet)
    - OSA & OSA/2
    - 3172
    - Adapters on P/390, R/390, Multiprise 3000
  - OSA Express on 9672 G5, G6 and zSeries 900
    - GigaBit Ethernet
  - Channel to Channel Adapter



# Additional Requirements



## LPAR/Basic Mode

- ECKD DASD
  - one 3380/3390/Multiprise Internal Disk
- Tape
  - One 3480/3490 drive
  - OS capable of writing to tape
- Character device
  - Hardware Maintenance Console (HMC)



# Additional Requirements



## VM Guest

- ECKD DASD
  - one 3380/3390/Multiprise Internal Disk or
  - one 600 cyl minidisk
- Character device
  - Virtual 3215 Console
- VM/ESA 2.4.0 or z/VM recommended



# Additional Support



## Processor Features

- Multiprocessing (SMP)
- IEEE floating point feature
  - emulated in software on machines without it
- Expanded Storage





# Additional Support Devices



- LPAR/Basic Mode
  - FBA DASD
    - 9336
- VM guest
  - FBA DASD
    - 9336
    - VDISKs
  - Network connections
    - Virtual CTCs
    - IUCV



# Marist College Joint-study Distribution



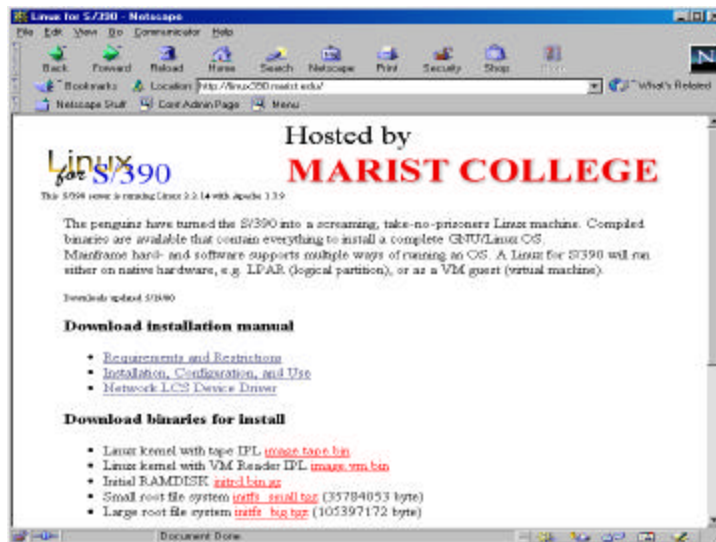
- Downloadable binaries
  - Pre-compiled Linux kernels
    - For IPL from tape (LPAR or basic mode)
    - For IPL from VM reader
  - Initial ramdisk file system
  - Small file system
  - Large file system
- Installation documentation



<http://linux390.marist.edu>



# Marist College Site



# Marist Distribution



- Operating environment
  - Same as "Cross-Compile"
- Support
  - Same as "Cross-Compile"
  - Linux-390 discussion list
    - Subscribe by sending an e-mail to [listserv@vm.marist.edu](mailto:listserv@vm.marist.edu) with the body of the text
      - ▶ "subscribe linux-390 your-name".



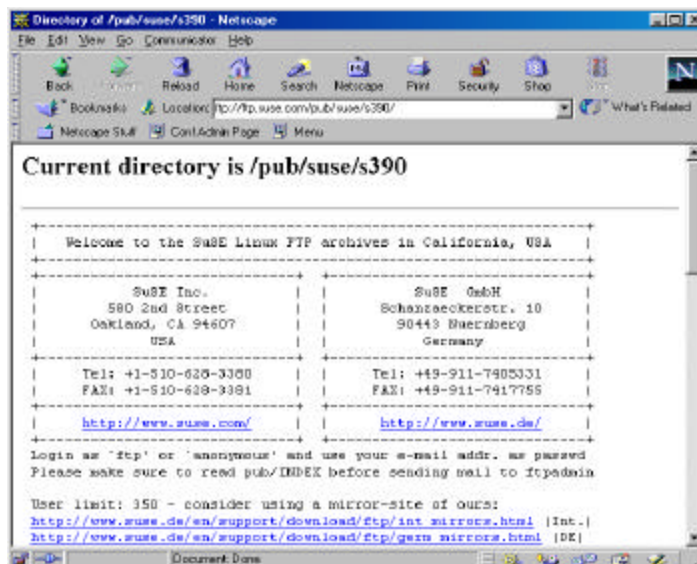
# SuSE Distribution



- Beta version available now from <ftp://ftp.suse.com/>
  - CD image of S/390 binaries
    - ▶ kernel
    - ▶ file system
    - ▶ over 400 applications
  - Installation documentation
- Informal support via Linux-390 list
- Requires workstation with CD-ROM



# SuSE FTP Site



# SuSE Distribution



- GA version due this fall
  - Formal support for
    - G5, G6, zSeries 900
    - Multiprise 3000
  - Support from
    - SuSE
    - IBM Global Services
  - Will require workstation with CD-ROM



# TurboLinux Distribution



- Beta version to be available soon
- GA planned for 4Q00
  - Formal support for
    - G5, G6, zSeries 900
    - Multiprise 3000
  - Support from
    - TurboLinux
    - IBM Global Services
- Watch <http://turbolinux.com>



# Applications



- Over 500 applications available from
  - <http://linux.s390.org/>



# Installing the Marist System



# Getting Started with the Marist Distribution



- Get some good books
  - Marist Install documentation  
<http://linux390.marist.edu>
  - Gordon Wolfe's cookbook  
<http://reason.marist.edu/cookbook.pdf>
  - Linux for S/390 Redbook  
<http://www.redbooks.ibm.com/pubs/pdfs/redbooks/sg244987.pdf>
- Enroll in the Linux-390 discussion list



# Installation Steps Marist Distribution



- Prepare the target environment
- Load the Linux kernel image and ramdisk file system into memory
- Build the Linux file system
- Build a boot disk
- Build a swap device



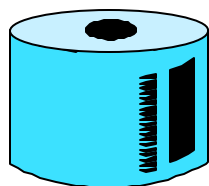
# Preparing the Target Environment (LPAR)



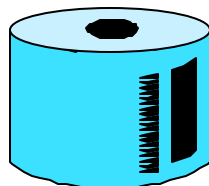
- Ensure IOCP allocates required resources
  - 2 3380/3390 devices
    - boot disk & root file system
    - swap volume
  - Network connection
    - OSA, OSA Express or CTC
  - At least 64MB central storage
- Make sure you know the addresses



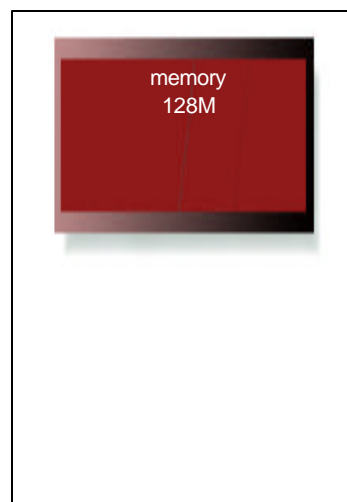
# Sample LPAR Configuration



300 dasd



301 dasd



interface  
(osa, etc)



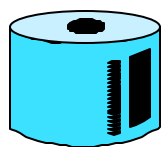
# Preparing the Target Environment (VM)



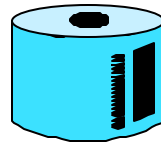
- Set up the virtual machine directory entry
  - 3 minidisks
    - 1000 cyl - root file system
    - 200 cyl - swap
    - 20 cyl - boot disk
  - Network connection to VM TCPIP
    - VCTCA or IUCV
  - At least 64MB of storage



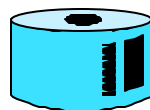
# Sample Linux Virtual Machine



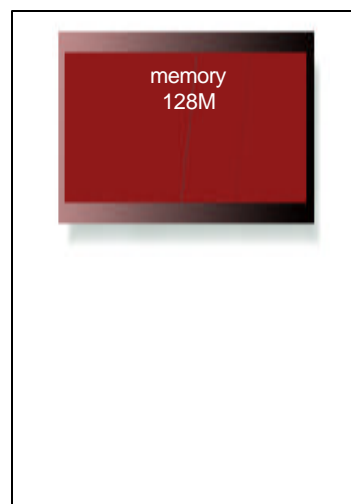
200 mdisk



201 mdisk



300 dasd





# Sample VM Directory Entry



```
USER LINUX01 LINUX01 128M 256M G
MACHINE ESA
IPL CMS
IUCV ANY PRIORITY MSGLIMIT 2000
IUCV ALLOW
CONSOLE 009 3215
SPOOL 00C 2540 READER *
SPOOL 00D 2540 PUNCH A
SPOOL 00E 1403 A
LINK MAINT 190 190 RR
LINK MAINT 19D 19D RR
LINK MAINT 19E 19E RR
MDISK 191 3390 nnnnn 0050 xxxxxxx MR
MDISK 200 3390 nnnnn 1000 xxxxxxx MR
MDISK 201 FB-512 VDISK 288000 MR
MDISK 300 3390 nnnnn 0020 xxxxxxx MR
```



# Prepare Linux Minidisks (VM)



- Format the minidisks using the CMS FORMAT command  
FORMAT 200 B  
FORMAT 201 C
- Reserve the minidisks with the CMS RESERVE command  
RESERVE 200 LINUX MDISK B6  
RESERVE 201 LINUX MDISK C6



# Update VM TCP/IP (VM)



- Update the PROFILE TCPIP for an IUCV link
  - Define the device and link

```
DEVICE DIUCVL01 IUCV 0 0 LINUX01 B
LINK LIUCVL01 IUCV 0 DIUCVL01
```
  - In the HOME section

```
v.v.v.v LIUCVL01 (vm side of link)
```
  - In the GATEWAY section

```
p.p.p.p = LIUCV01 1500 HOST (Linux side)
```
  - For automatic start of connection

```
START DIUCVL01
```



# Gather the Network Parameters (VM or LPAR)



- |   |  |
|---|--|
| <ul style="list-style-type: none"><li>■ For a multi-host connection<ul style="list-style-type: none"><li>— host name</li><li>— IP address</li><li>— network mask</li><li>— network address</li><li>— broadcast address</li><li>— gateway address</li><li>— IP address of DNS server</li><li>— DSN search domain</li></ul></li></ul> | <ul style="list-style-type: none"><li>■ For a point to point connection<ul style="list-style-type: none"><li>— host name</li><li>— IP address</li><li>— network mask</li><li>— network address</li><li>— peer address</li><li>— gateway address</li><li>— IP address of DNS server</li><li>— DSN search domain</li></ul></li></ul> |
|---|--|



## Obtain the Linux Code (LPAR)



- Transfer the Linux components to a local system using a blocksize of F 1024
  - via FTP from linux390.marist.edu
  - via the web site at <http://linux390.marist.edu>
- kernel image
  - ▶ image.tape.bin
- ram disk
  - ▶ initrd.bin
- Large File System
  - ▶ initfs\_big.tgz



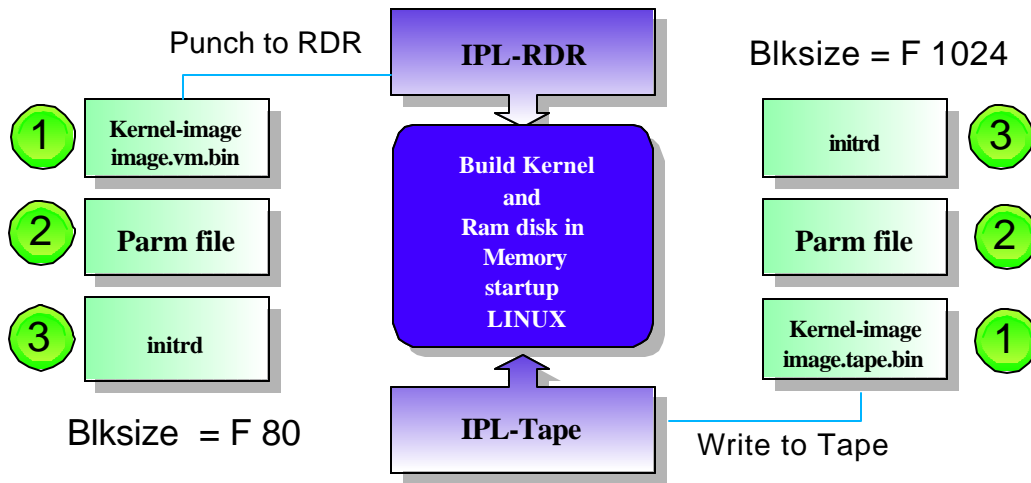
## Obtain the Linux Code (VM)



- Transfer the Linux components to your VM system using blocksize of F 80
  - via FTP from linux390.marist.edu
  - from the web site at <http://linux390.marist.edu>
- kernel image
  - ▶ image.vm.bin --> image\_vm txt
- ram disk
  - ▶ initrd.bin --> initrd txt
- Large File System
  - ▶ initfs\_big.tgz --> initfs\_big tgz



# Initial System Build Overview



# Create the Parm File

## ■ LPAR

```
mem=128m
dasd=300,301
root=/dev/ram0 ro
```

## ■ VM

```
mem=128m
mdisk=200,201
dasd=300
iucv=TCPIP
root=/dev/ram0 ro
```



## Load Linux from Tape (LPAR)



- Copy 3 files to tape with DITTO, CMS MOVEFILE or IEBGENR (ensure blksize F 1024)
  - image.tape
  - parm file
  - initrd
- IPL from the tape



## Load Linux from VM Reader (VM)



- Punch the kernel, the parm file and the ramdisk to the reader

```
SP PUN * CLOSE
PUNCH IMAGE_VM TXT A (NOH
PUNCH INIT PARM A (NOH
PUNCH INITRD TXT A (NOH
```
- IPL from the reader

```
IPL 00C CLEAR
```



# Bring Up the Linux System with the Ramdisk File System (VM or LPAR)



- When prompted
  - Are you connected to the network?
    - Reply yes
  - Enter the kind of network
    - select CTC (if using VCTC or IUCV)
    - select appropriate network (if in LPAR)
    - enter the other network parameters
  - Give root password for maintenance
    - enter pass4root



# Establish the IUCV Connection (VM)



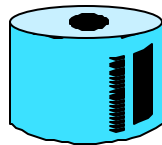
- Enter

```
ifconfig iucv0 p.p.p.p pointopoint v.v.v.v mtu 1500
route add -net default iucv0
```

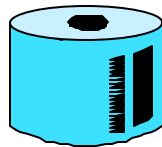
  - where
    - ▶ p.p.p.p is the IP address of the Linux end of the connection
    - ▶ v.v.v.v is the IP address of the VM TCP/IP end of the connection



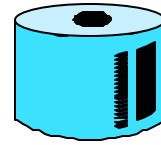
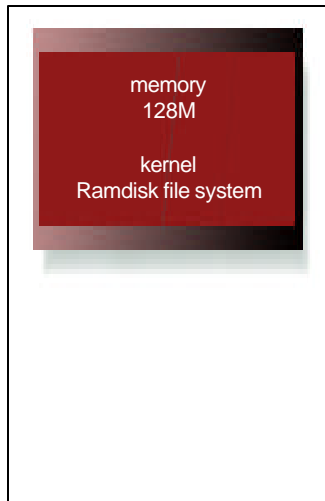
# Initial Linux Environment (LPAR and VM)



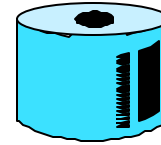
300 dasd  
dasda



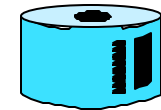
301 dasd  
dasdb



200 mdisk  
mnda



201 mdisk  
mndb



300 dasd  
dasda



# Build the File System (LPAR)



- Create and mount an empty file system

```
dasdfmt -f /dev/dasda -b 4096
```

```
mke2fs /dev/dasda1 -b 4096
```

```
mount -t ext2 /dev/dasda1 /mnt
```

- Move the file system tar file to the Linux system

```
cd /mnt
```

```
mkdir tarpit
```

```
ftp to the Linux system
```

```
binary
```

```
get initfs_big.tgz /mnt/tarpit/initfs_big.tgz
```

```
tar -xzvf /mnt/tarpit/initfs_big.tgz
```



# Build the File System (VM)



- Create and mount an empty file system

```
mke2fs /dev/mnda -b 4096
mount -t ext2 /dev/mnda /mnt
```

- Move the file system tar file to the Linux system

```
cd /mnt
mkdir tarpit
ftp to the Linux system
binary
get initfs_big.tgz /mnt/tarpit/initfs_big.tgz
tar -xzvf /mnt/tarpit/initfs_big.tgz
```



# Update the FSTAB File (VM)



```
cd /mnt/etc
cp fstab fstab.save
ed fstab
1
.c
/dev/mnda /ext2 defaults,errors=remount-ro 0 1
.
1,$p /dev/mnda / ext2 defaults,errors=remount-ro 0 1
1,$w none /proc proc defaults 0 0
q
```





# Prepare an IPLable DASD (VM)



- Create a file system on the boot device

```
dasdfmt -f /dev/dasda -b 4096
```

```
mke2fs /dev/dasda1 -b 4096
```

- Move boot files to the boot disk

```
mount -t ext2 /dev/dasda1 /tmp
```

```
cd /tmp
```

```
mkdir boot
```

```
cd /boot
```

```
cp * /tmp/boot
```



# Prepare an IPLable DASD ... (VM)



- FTP the kernel image to the boot directory

```
cd /tmp/boot
```

```
ftp xxxxxx
```

```
binary
```

```
get image_vm txt image.vm.bin
```

```
quit
```

- Create a parm file in the boot directory

```
ed image.vm.parm
```

```
.a
```

```
mem=128m mdisk=200,201 dasd=300 iucv=TCPIP
```

```
root=/dev/mnda ro noinitrd
```

```
.
```

```
1,$p
```

```
1,$w
```

```
q
```



# Prepare an IPLable DASD (LPAR)



- FTP the kernel image to /boot

```
cd /mnt/boot
ftp xxxxx
binary
get image.tape.bin
quit
```

- Create a parm file in /boot

```
ed image.tape.parm
.a
mem=128m dasd=300,301 root=/dev/dasda ro noinitrd
.
1,$p
1,$w
q
```



# Create the IPL Records & Reboot (LPAR & VM)



- LPAR

```
silo -f image.tape.bin -d /dev/dasda -p
image.tape.parm -b ipleckd.boot -t2
shutdown -h now
IPL from device 300
```

- VM

```
silo -f image.vm.bin -d /dev/dasda -p
image.vm.parm -b ipleckd.boot -t2
shutdown -h now
IPL 300 CLEAR
```

- Both LPAR & VM

```
enter network parameters (again)
enter root password
```



# Establish the IUCV Connection (VM)



## ■ Enter

```
ifconfig iucv0 p.p.p.p pointopoint v.v.v.v mtu 1500  
route add -net default iucv0
```

### ● where

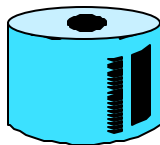
- ▶ p.p.p.p is the IP address of the linux end of the connection
- ▶ v.v.v.v is the IP address of the VM TCP/IP end of the connection



# After Build of File System (LPAR and VM)

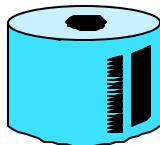


boot device

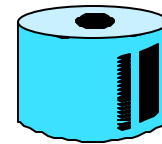
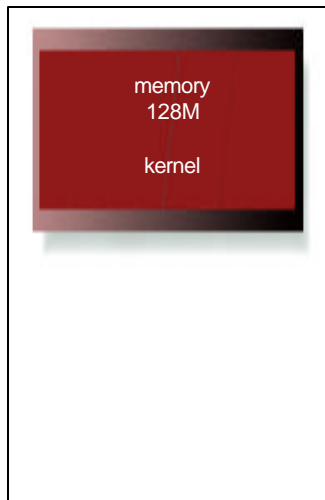


large file system

300 dasd  
dasda

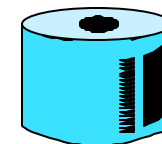


301 dasd  
dasdb

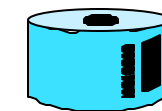


large file system

200 mdisk  
mnda



201 mdisk  
mndb



boot device

300 dasd  
dasda



# Create a Swap Device (LPAR)



- Create the swap space

```
dasdfmt -f /dev/dasdb -b 4096
mkswap /dev/dasdb
swapon /dev/dasdb
chmod 0600 /dev/dasdb
```

- Update the FSTAB file

```
cd /etc
cp fstab fstab.save2
ed fstab
1
.i
/dev/dasdb swap swap defaults
```

```
.
1,$p /dev/dasdb swap swap defaults
1,$w /dev/dasda / ext2 defaults,errors=remount-ro 0 1
q none /proc proc defaults 0 0
```



# Create a Swap Device (VM)



- Create the swap space

```
mkswap /dev/mndb
swapon /dev/mndb
chmod 0600 /dev/mndb
```

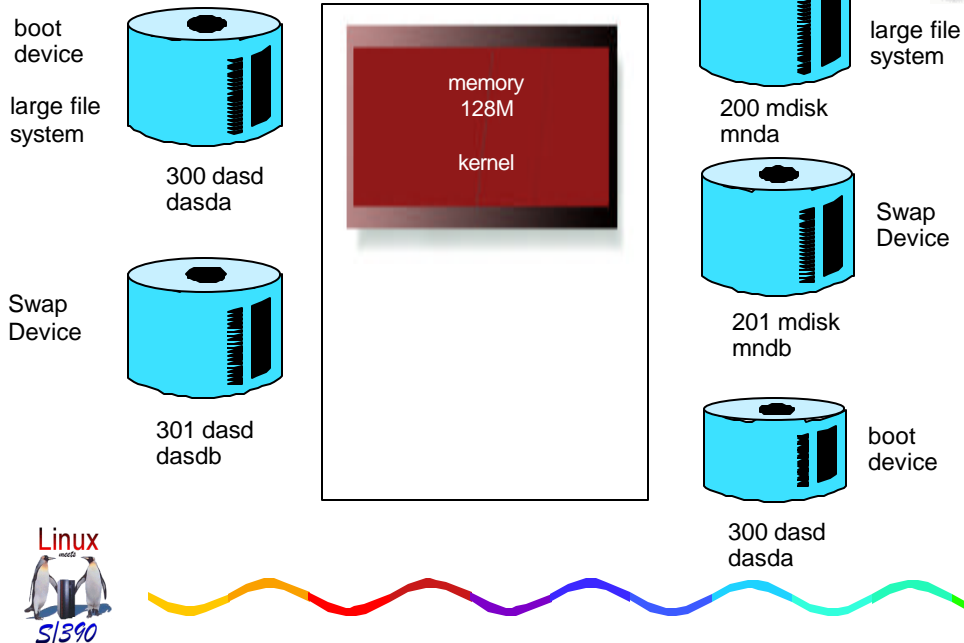
- Update the FSTAB file

```
cd /etc
cp fstab fstab.save2
ed fstab
1
.i
/dev/mndb swap swap defaults
```

```
.
1,$p /dev/mndb swap swap defaults
1,$w /dev/mnda / ext2 defaults,errors=remount-ro 0 1
q none /proc proc defaults 0 0
```



# After Swap Device Creation (LPAR and VM)



# Automate Start of IUCV (VM)

- Create an ifcfg-iucv0 file

```
cd /etc/sysconfig/network-scripts
```

```
mv ifcfg-ctc0 ifcfg-iucv0
```

```
ed ifcfg-iucv0
```

```
1
```

```
.c
```

```
DEVICE=iucv0
```

```
.
```

```
1,$p
```

```
1,$w
```

```
q
```

```
DEVICE=iucv0
USERCTL=no
ONBOOT=yes
BOOTPROTO=none
REMIP=9.82.56.1
NETWORK=9.82.56.0
NETMASK=255.255.255.0
IPADDR=9.82.56.131
```



# Automate Start of IUCV ... (VM)



- Edit the network script

```
cd /etc/sysconfig
ed network
4
.c
GATEWAYDEV=iucv0
.
1,$p
1,$w
q
```

```
NETWORKING=yes
FORWARD_IPV4=no
HOSTNAME=linux01
GATEWAYDEV=iucv0
GATEWAY=9.82.56.1
```



# Automate Start of IUCV ... (VM)



- Create an ifup-local file in /sbin
- Make it executable

```
cd /sbin
chmod +x ifup-local
```

```
#!/bin/sh
if [ "$1" != "lo" ]
then
    ifconfig $1 mtu 1500
fi
```

