

~~1901~~

BASIS UNIX NOTES

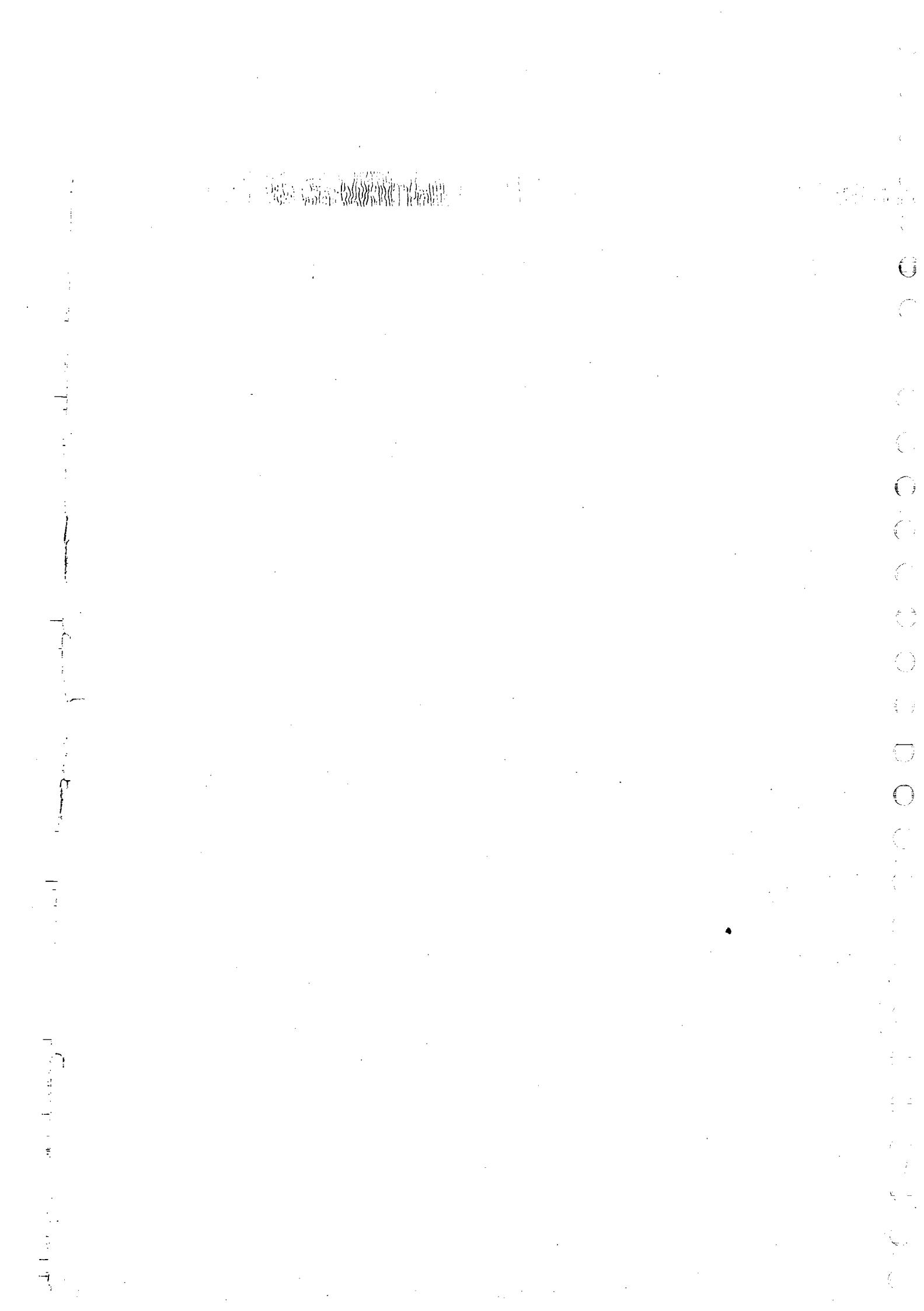
(WST)

SAI XEROX

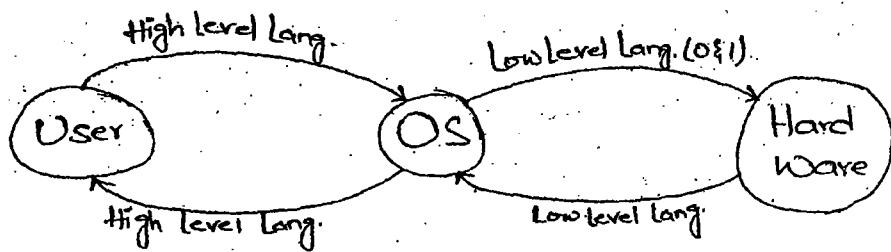
AMEERPET,

HYD

9885440924



BASIC UNIX



Operating System is divided into 2 parts.

- Single User OS:

only one Person can access all the System Resources. Eg: Win 95, Dos

- Multi User OS:

More than one Person can access all the System Resources at a time.

Eg: Win 2000, XP, Vista.

Unix flavors :-

* Solaris

* Aix

* HP-UX

* Linux

} All these are Multi User OS.

In Multi-user Environment Terminal comes into Picture.
(Users workspace is called Terminal).

Terminal

User's Workspace is called Terminal.

- 1. Smart Terminal: It has Full System Resources.

Eg: ~~Monitor~~, CPU.

- 2. Dumb Terminal: It does not have full System Resources.

Eg: Monitor, Keyboard & Mouse.

It is also called as Server. (Which Provide Services).

History of UNIX:

- In 1958 R&D Started on OS.
- In 1961 Ken Thompson who belongs to AT&T Bell labs introduced Single User OS in "B" language (binary lang) on PDP-7 Machine. (Programmable Database Processor).
- In 1971 Dennis Ritchie & Ken Thompson introduced Multiuser OS in "C" language on PDP-11 Machine.
- After 1972 Unix Source Code became Free Source Code.

Vendors

BSD (Berkeley Software Distribution)

HP (Hewlett Packard)

IBM (Intl. Bus. Machine)

Sun Micro Systems

RHEL

(Red Hat Enterprise Linux).

Flavors

NET.BSD O/S [designed for his company. It's failed]

✓ HP-UX O/S

✓ AIX O/S
(Advanced Interactive Execution)

✓ Solaris O/S.

✓ Linux O/S

HP-UX latest Version is HP-UX 11.xi

AIX latest Version is 6.1

Solaris latest Version is 5.10

Linux latest Version is 5.0

Features of UNIX:

1. Multiuser
2. Multi Tasking
3. Security
4. Help
5. Communication
6. Utilities
7. Programming Facility.

GUI - Graphical User Interface.

CLI - Command line Interface.

1. MultiUser:

Difference :-

- Windows Operating System Supports 30,000 Users at a time.
- Unix OS Supports 2 million Users at a time (depending on System Architecture).

2. Multi Tasking :-

Difference:

- In Windows System May hang if we open 10-15 tasks.
- In Unix we can send 400 tasks to background.

In Unix we have 2 Process

Fg (Foreground Process)

Bg (Background Process)

3. Security :

- In Unix we have login level security and FAP Security.
Login level means ^{at} Password

- In windows it shows password strength.
(* * *) By that we can hack.

But

In Unix it won't show password strength.
Cursor remains at same place.

- In windows by default we get execute permission of a file.
Where as in Unix we won't get any execute permission by default.

4. Help : → MAN
 → HELP
 → --info } Using these Commands we can get help

Eg: Man ls <

No diff.

5. Communications:

We have 2 types of Communications

(i) Online Communication → Write
 → Ytalk

(ii) Offline Communication → Mail

Write - (only one way Communication)

Ytalk - (two way Communication, Eg: YMessenger)

Write and Ytalk is Normal User Commands.

Mail is administrator Command. As a normal user we cannot use Mail Command.

\$ - Normal User.

- Administrator.

Mail is a Command we can send offline messages.

6. Utilities: (Commands)

- In Windows OS we won't get more than 400 Commands.
- In Unix OS we get More than 400 Commands.

7. Programming facility:

- When Installing Unix Os it Supports 50-60 languages.
- In windows by default it won't support any languages.

Important

Vi editor
FAP
Find

File System of UNIX:

Organized files & directories in efficient manner is called File System. (Collection of files & dir.).

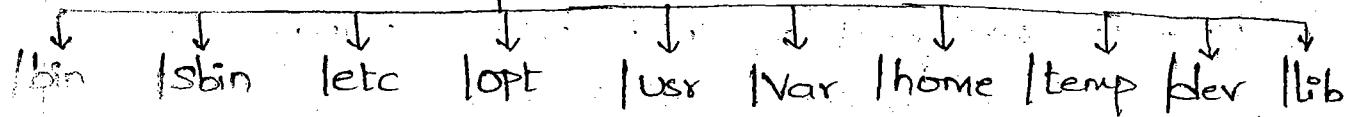
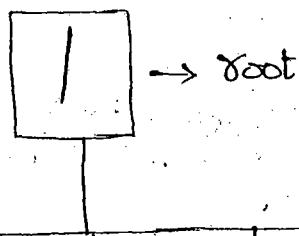
Directory - Collection of files and Subdirectories.

File - Collection of Records.

<u>Windows</u>	<u>Linux</u>	<u>HP-UX, AIX</u>	<u>Solaris</u>
FAT 16	EXT2	JFS	UFS
FAT 32	EXT3	(Journal Filesys)	(Unix file system)
NTFS	(extended fs)		
(New Tech.)			

+ Top Most directory in Unix is Root.

Symbol:



/bin: Directory Contains all Normal User Commands.

Eg: Cat, ls, mkdir

/sbin: Directory Contains all Administrator Commands and also Normal User Commands.

Eg: Wall, Useradd, Userdel

/etc: Contains all System Configuration files.

Eg: /etc/passwd - Contains all user information.

/etc/shadow - Contains all user password information.

/opt: Contains all third party Software information.

/usr: Contains all System messages and also contains 90% of OS.

/var: Contains logs information.

/home: Contains all user names.

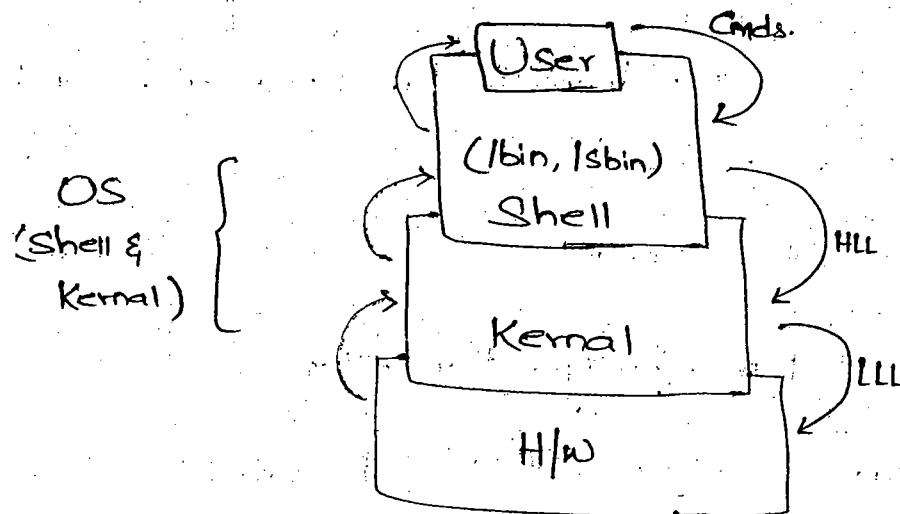
/tmp: Contains all temporary files.

/dev: Contains all logical names of physical devices.

/lib: Contains all library files.

[for 3rd Party SW we need Compilers. Compilers are under /lib files].

Architecture of UNIX:



Shell: Shell is an interface between User & Kernel.

Kernel: Kernel is an interface between Shell & H/w.

- Shell is a Command checker. Internally it Maintains lbin, lsbin.

Functioning: User Sends Commands to Shell.

Shell Searches them in lbin & lsbin directory, then it Sends to Kernel if they exist.

Kernel Converts HLL to LLL and Sends to H/w. After Execution Vice Versa.

Login:

Unix <

: Login : Sunman095 <

[Sunman095 @ unixlab Sunman095] \$
 | |
 | Username Node name | Directory
 | |

To Set Password, Command is **Passwd**

Rule: - Password Should be Min: 6 char. & Max: 14 char.

- Password Should be Combination of numbers & chars.
- We can use Special char. also.

Ex: \$ **Passwd** ↵

Type: ↵

Re-Type: ↵

To logout

\$ **exit** ↵ (or) \$ **Logout** ↵ (or) **Ctrl+D**.

* Unix OS follows 7 types of files:

1. Normal file (-)
2. Hidden file (.)
3. Directory file (d)
4. Backup File (~)
5. Linked file (@)
6. Executable file (*)
7. Device file () -no symbol.

- To create a file in Unix OS we have 4 Commands.

(i) **Cat**

(ii) **Touch**

(iii) **Vi**

(iv) **Tee**

I. Normal file:

To Create file Using Cat:

Syntax:

`$ Cat <option> <filename>`

-n → assign no: to records.

-b → assign no: to records wherever data records are there.

-s → Squeeze. It will Squeeze Multiple blank lines into Single blank line.

Eg: \$ Cat > Raj <

Hi this is Raj.

Ctrl+d (Save & quit from a file).

To View a file.

Eg: \$ Cat < Raj <

Hi this is Raj.

\$ Cat -n Raj <
1 —
2 —
3 —
4 —

\$ Cat -b Raj <
1 —
2 —
3 —
4 —

\$ Cat -s Raj <
1 —
2 —
3 —
4 —

\$ Cat -sn Raj <

\$ Cat -n > <filenames> < gives no: while creating a file.

To add data in an existing file. (Appending data).

\$ Cat > Raj <

Ctrl+d <

Touch:

\$ touch <filenames> <

- Using touch we can create an empty file. We cannot enter data.

- Using touch we can create multiple files at a time.

\$ touch file1 file2 file3 <

- In Cat we cannot create multiple files.

ls: It will list out all files & directories from the Present Working Directory (PWD).

Eg: \$ ls <

3. Hidden Files (.) :

Eg: \$ Cat > .Raj <

To view hidden files

\$ ls -a <

- It will show all hidden files & dir. and also normal files & dir.

Empty hidden file.

\$ touch .file1 <

3. Directory File (d):

To Create a directory we use **Mkdir**

```
$ mkdir <options> <dir.name>
```

- P → Make Parent directories.
- M → assign desired Permissions.

Eg: \$mkdir dir1 ↴

In Unix, in the same location we cannot create directory with the same name.

- we can create multiple directories at a time.

Eg: \$ mkdir dir1 dir2 ↴

to view

```
$ ls ↴
```

To Create directory under directory.

```
$ mkdir -P d1|d2|d3|d4 ↴
```

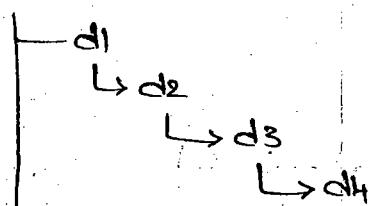
```
$ ls ↴
```

d1

to see sublevel info also.

```
$ tree ↴
```

- Shows tree structured format.



to see only under d2:

```
$ tree d1|d2 ↴
```

\$Clear: is used to clear the file.

Multilevel Directories:

\$mkdir -p d1|d2|d3|d4 ↴

\$tree - to view tree Structure.

Changing Directory:

. → Present Working Directory

.. → Parent Directory

~ → Home Directory

/ → Root Directory

Syntax:

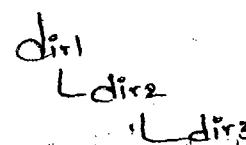
\$cd <dir.Name> ↴

Eg: \$cd dir1 ↴

Sub.dir \$cd dir1|dir2|dir3 ↴

To go two steps backward

\$cd ... ↴



Being in dir1 we can Create file in dir3.

\$cat > dir1|dir2|dir3|apple ↴

(or)

\$cd dir1|dir2|dir3 ↴

\$cat > apple ↴

Being in dir3 we can create file in Parent Directory

dir3]\$ cat > ..|file1|

Under Home directory

dir3]\$ cat > ~|file2|

To view files in home directory being in dir3

dir3]\$ ls ~|

To Create in other directory

dir3]\$ touch ~|d2|d3|d4|file1|

\$ Pwd | - Show you Present Working directory with Path.

Eg: \$ Pwd |

|Users|sunmanos|dir1|dir2|dir3

Absolute Path.

Removing files:-

Syntax:

\$rm Option > <filename>

-i → Interactive (Confirmation)

-v → Verbose (After removal, it will give removed)

-f → forcefully

Eg: \$ rm file1 ↴

\$ rm file1 file2 file3 ↴ Can Remove Multiple files at a time.

\$ rm dir1|dir2|dir3|apple ↴

\$ rm ~|f1 ↴

being in directory3 how can we remove multiple files in home directory.

\$ rm ~|{file1, file2, file3} ↴

\$ rm ..|..|file1 ↴

\$ rm ~|dir1|file1 ↴

Removing Directories:

\$ rm <options> <dirname> ↴

-r → Recursively

-v → Verbose

-i → Confirmation

Eg: \$ rm -r di ↴ (Removes all files & Sub directories in a directory & also directory)

To Remove Empty Directories

\$ rmdir <Empty dirname> ↴

Multiple Directories:

\$ rm -r dir1 dire ↴

Copying files:-

Syntax:-

\$ CP <option> <Source> <destination>

-i → interactive

-v → Verbose

-b → backup file.

file → file

file → Dir

Dir → Dir

In Copy Command we have 2 Methods.

1. Absolute Method: The Path Starts from the Root level onwards. is Called Absolute Method.

2. Relative Method: The Path starts from the Pwd. is called Relative Method.

Eg: Using Absolute Method

\$ CP /users/Sunman095/f1 /users/Sunman095/f2 ↵

Eg: Using Relative Method

\$ CP f1 f2 ↵

Absolute Method is useful Copying files from one system to another.

Eg: \$CP /users/Sun095/apple /users/Sun096 ↵

Verbose Means it will just pass a message.

Eg: \$CP -b f1 f2 ↵

f₂ will have all f₁ Content & then f₁ will be copied into f₂.

f₂ file will ^{be} automatically created.

File → Dir.

\$cp apple d₁/d₂/d₃ ↘

\$cp s1 ~apple ↘

Dir → Dir

\$cp <option> <Source> <destination>

-r → recursive

-i

-v

Eg: \$cp -r d₁ d₂ ↘

Moving files:-

Syntax:

\$mv <options> <source> <destination>

-i

-v

-b

Eg: \$mv f₁ f₂ ↘

\$mv /users/Sun095/f₁ /users/Sun095/f₂ ↘

By Using \$mv Command We Can Rename filename & also directory name.

Eg: \$mv f₂ f₉ ↘

\$mv d₁/d₂/d₃ d₁/d₂/d₄ ↘

Moving Directories :-

`$mv <option> <source> <destination>`

-i

-v

Eg: `$mv d1/d2 d1/ ↲`

`$mv /Users/sun095/d1/d2 /Users/sun096 ↲`

To add the data of a file to another file without erasing the data of that file.

`Cat f1 >> f2 ↲`

Wild Characters :-

There are 3 types of wild characters in Unix OS.

- (i) ? → indicates anyone character [0-9] or [A-z]
- (ii) * → indicates all characters
- (iii) [] Range.

Purpose of Wild Characters is we can search files and directories in shortcut way.

(i)

Eg: `$ ls ? ↲`

displays all single character files & directories.

`$ ls ?? ↲`

displays all double character files & directories.

for directory it displays till the firstlevel file (or) dir.

i.e., d1
↳FFF

(ii) \$ ls *

displays all the files & dir. with all characters.

Eg: \$ ls f*

OP: ff

\$ ls d*

To display all 3 character files.

\$ tree ***

Eg: \$ cp ? dl

all the single character files will be copied into dl.

\$ cp -i ? dl

asks Confirmation for each file.

Eg: \$ cat ?

displays the content of single char. files.

\$ cp -r ? dl

Eg: \$ ls f*

(iii) \$ ls [a-z] it is case sensitive.

Single bracket indicates Single char.

\$ ls [a-z]

\$ ls [a-b]

Eg: cat [a-zA-Z]

display all single char. file contents.

Eg: \$ ls [a-z][0-9] ↴

Eg: \$ rm di/* ↴

Remove all files of di dir.

\$ rm -i di/* ↴

Options of ls :-

i. ls -a : (displays all hidden files & normal files)

ii. ls -d : (all directories)

iii. ls -F : (Show differentiate b/w files & dir.)

iv. ls -i : (show you inode no: of files & dir.)

v. ls -l : (gives longlist format of files & dir.)

vi. ls -n : (give you groupid & userid of users)

vii. ls -R : (Recursively it searches for files & dir.)

viii. ls --full-time : (gives longlist format. Same as ls-l here it shows year also),

ix. ls -s : (Shows you default size of file & dir.).

Note:

Default size of file & dir. at the time of creation is 4KB (4096 bytes).

If any file ended with * i.e., executable file.

ls -l Shows 9 fields of output.

1st field indicates permission of files (or) directories.

2nd field indicates links of files (or) dir.

3rd field indicates owner of files (or) dir.

- 4th field indicates group information.
- 5th field indicates Current size of file in bytes.
- 6th field indicates Month info.
- 7th field indicates Creation of Date } last Modified.
- 8th field indicates Creation time.
- 9th field indicates Name of file (or) dir.

Eg: ls -nl

it shows groupid & userid and format

ls -ld dir3 <

Shows longlist format of dir3.

Re- Directions: we can store any command output in a particular file.

We can make it permanent also.

3 types:

1. Output Redirection (> or |>)
2. Input Redirection (<)
3. Error Redirection (2>)

1. Output Redirection: (> or |>)

Eg: \$ cat -n apple > f1 <

to put no: permanently

Redirection is Possible b/w only file to file.

\$ ls > f1 <

\$ ls >> f1 <

2. Input Redirection: (In simple filter class).

3. Error Redirection:

\$ cat * 2> f9 <

all errors will be there in f9.

Eg: cat * 1> f10 <

Note:

> - output is copied into a file. Eg: cat * > f1 <

>> - only error msgs. will be displayed. Content will be copied into file. Eg: cat * >> f1 <

2> - displays contents. error msgs. will be in file.

Eg: cat * 2> f1 <

>> - it will append the result to the file.

Eg: cat * >> f1 <

Vi editor :- (only files)

Syntax:

\$ Vi <option> <filename>

- O (Open Multiple files at a time).

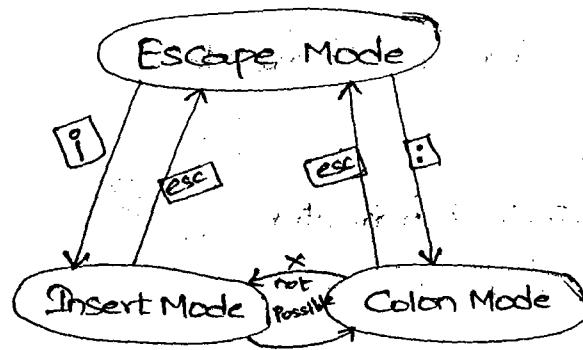
It follows 3 Modes.

(i) ESCape Mode (esc) (ii) Insert Mode (i) (iii) Colon Mode (:)

- Cursor Moving
- Delete the data
- Yank (or) Copy data
- Paste data
- Replace the data

- We can only insert data.

- Save the file
- Quit
- Save & Quit
- Set the no:



Cursor Moving Options:

l - Cursor will Move to one char Right.

h - one char left.

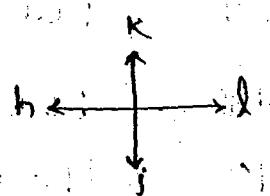
j - one line down

K - one line up

w - one word forward

b - one word backward

5w - five words forward



5b - five words backward
\$ - end of line
^ - Starting of the line
5j - five lines down
5k - five lines up
M - Cursor will go to Middle of the Page.
L - Cursor will go to end of Page.
H - Starting of Page.
G - End of the file.
gg - Starting of the file.

Deleting options :-

x - delete Current Cursor Character.
u - Undo
X - deleting Previous Character
- - redo
dw - delete the Current Cursor word.
5dw - delete the 5words from Cursor Position
db - delete the Previous word
5db - delete the 5 words (5 Previous) from Cursor Position.
d\$ - delete till the end of line from Cursor Position
d^ - delete "till" the Start of line from Cursor Position.
5dj - delete 5 lines forward from Cursor
5dk - delete 5 lines backward from Cursor.
dd - delete Current Cursor line.
d1 - delete till Middle of the Page.
d! - delete till end of the Page.

dH - delete "" starting of The Page

* dG - delete till End of the file.

dgg - delete till Starting of the file.

Copying Options :-

yw - Copy Current Cursor Word

P - Pasting

5yw - Copy 5 words

yb - Copy Previous Word

5yb - Copy 5 words backward

y\$ - Copy till end of line

y^ - Copy till Starting of line.

5yj - Copy five lines forward

5y~~k~~ - Copy 5 lines ~~backward~~ up.

yy - Copy Current Cursor line.

yM - Copy till Middle of Page

YL - Copy till end of Page

yH - Copy upto Starting of Page.

yG - Copy till end of file

ygg - Copy upto Starting of file.

Replacing Options :-

r - Replacing Current Cursor Char.

R - Replacing entire line.

ii) Insert Mode:

i → to insert data.

iii) Colon Mode:

:w → Saving data.

:q → Quitting without saving

:wq! → Save, quit & forcefully quit

: Set nu → Setting no:

: Set nonu → Removing no:

:40 → to goto 40th line

:%s |hai| ttt|

(Substituting hai with ttt).

To open multiple files at a time in Vi:

\$ Vi -o f1 f2 f3 ↵

Ctrl + w → to move to next file.

-o works only in linux os.

File Access Permissions : (FAP)

c - Character Spl. file
b - block Spl. file

\$ ls -l <.

We get long list format. In that first field is Permission field.

Character Special file :- (c)

It Maintains Physical paths of physical devices.

Block Special file :- (b)

It Maintains logical Paths of Physical devices. We can insert data here. (only admin can insert).

In this Permission field we have 10 Columns.

filetype [user group others]

Members

Permission

Values

Users → u

Read - r

→ 4

Write - w

→ 2

Execute - x

→ 1

rwx-7

Group → g

Read - r

Write - w

Execute - x

Others → o

Read - r

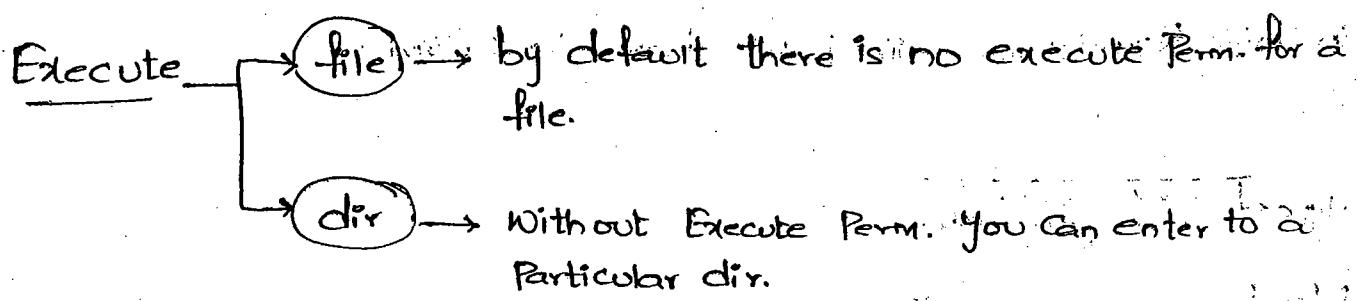
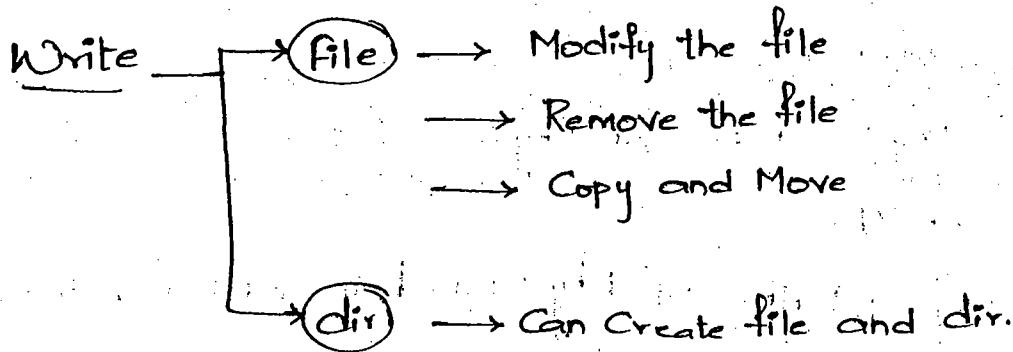
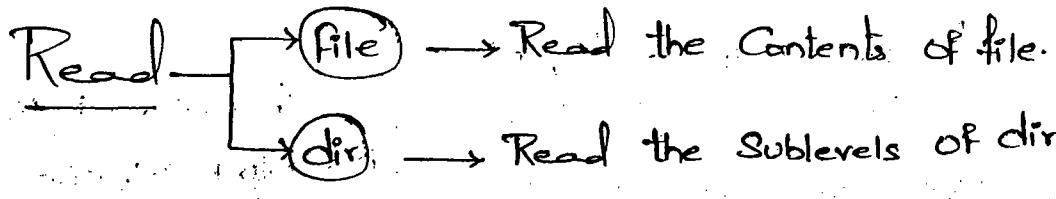
Write - w

Execute - x

User → Who Creates file.

Group → belongs to same dept. of the creator of file.

Others → belongs to other dept.



Max. Perm

user	group	others
6	6	6
Rw-	Rw-	Rw-

Dir

7	7	7
Rwx	Rwx	Rwx

Default

6	4	4
Rw-	R--	R--

7

Rwx	R-x	R-x
-----	-----	-----

\$ UMASK (User Mask Value) (it's a hidden value).

default 022

Subtract value from Max. Perm & default value.

To Change Umask Value

\$ umask <new value>

- Whenever you give Odd Value to Umask then Kernel will add one More Value (Perm) to file.

i.e.,

\$ umask 111 ↲

$$\begin{array}{r} 666 \\ -111 \\ \hline 555 \\ +1+1+1 \\ \hline \underline{\underline{666}} \end{array}$$

This is only for file. not for directories.

- we can give odd umask value in directories.

- When we give execute permission to a file, it acts like a command.

- To execute a file

\$. <filename> ↲

To Change Umask Value
Permanently

\$ vi .bash_profile

goto last line in file and
type umask <newvalues>
Save & exit. It will be changed

- If it doesn't change Logout & login
type \$umask
you will see changed value.

Three Ways of Providing Permission for files & dir:

1. Before Creating files | dir. → \$umask

2. After Creating files | dir. → \$chmod

3. At the time of Creating dir.

3. At The time of Creating dir.

Ex: \$ mkdir -m 777 dir1 ↴

↓

To specify Mode of Perm.

Ex: \$ mkdir -Pm 700 d1|d2|d3|d4|d5

2. After Creating files | dir.

Absolute Method / Symbolic Way

chmod.

Numerical Way

Symbolic Way

+ → add Permission

- → Remove Permission.

Ex: 1. \$ chmod g+s, o-s fi ↴

g+s

2. \$ chmod ugo+rw fi ↴

a=rw fi ↴

a → all the members.

Numerical Way

\$ chmod 600 fi ↴

Giving Permission to multiple dir.

\$ chmod -R 700 d1|d2|d3|d4

Insert Options :-

a - Cursor to goto next character place.

o

If we change anything in ".bash-Profile" file, and if you want it to reflect then you need to execute that file.

To execute

• fi

sh .fi

• fi

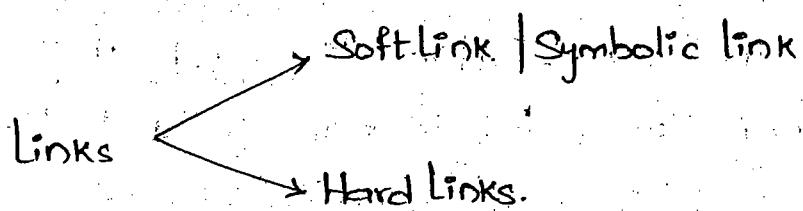
- To give Permission With in Users we need to give full Permissions of home directory.

\$ chmod 777 ~

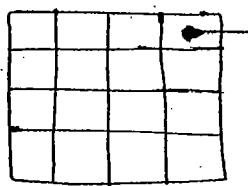
① When U1 Copy file from U2 then U1 will have Permission to change that file.

② When U1 sends file from U1 → U2 then U2 will not have any Permissions except reading.

LINKS :- is a Shortcut way to access files and directories.



Hard disk is divided into Several blocks



In linux each block Contains 1024 bytes.

Linux - 1024 bytes.

Solaris - 512 bytes.

In Unix Kernel will identify a file by its Inode no.
inode no. is divided into 2 parts.

- file name Part (FNP)

- Data Part (DP) - contents of file.

FNP Stores fileType, fileSize, Perm. of the file, uid, gid, Path of file.

To Create a Soft link

```
$ ln -s ~|d1|d2|d3|d4|file1 ~
```

```
$ ln <option> <Source> <destination>  
-s (Softlink)
```

- When a Softlink is Created the Inode no. of a Softlink file will change.
- It acts like Pointer which locates Original file. It just has Softlink file & Inode no.. There will not be any data.

Advantages:

1. Multiple Softlinks can be Created.
2. Creating Softlink file → file.
3. Creating Softlink dir → dir.

Disadvantage:

1. If one Source is removed, you can't access Softlink file.
- The Softlink file has Path in it.
- When a Softlink file is Created it shows its Permission as 777. We Cannot Change its Permissions. When you try to change it will change Source file Permissions.
- Here it displays 777 at any cost but if any other user want to modify he cannot. They can read that's all.

Hard Link :-

```
$ln <Source> <destination>
```

- There will not be any option in hard link.

Eg: \$ln ~|d1|d2|d3|d4|file1 ~

Hard link file doesn't occupy any space in hard disk.

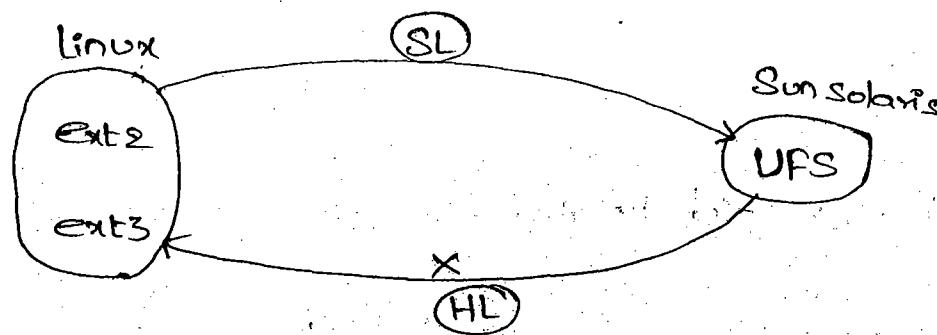
- Inode no. will be same.
- The size will be same.
- When you remove hard link the datapart will be safe.
(source).
- In softlink if you remove source everything will be gone.
- If source is removed still we can access datapart in hard link.

Advantages:-

- Multiple hard links can be created.
- If source is removed, then also you can access Hard link file.

Disadvantage:-

- hardlink from dir \rightarrow dir is not possible.



not Possible because we cannot access one filesystem with another filesystem.

Directory is nothing but a file system.

FIND Command:-

Syntax:-

```
$ find <Path> <options>
```

...
~
/ ...
-type
-name
-Empty
-Size
-Perm
-Inum

-links
-atime
-mtime
-Maxdepth
-Mindepth
-exec { } 1;
-ok { } 1;

closing syntax.

Options:

-type
↳ f - normal file
↳ d - directory
↳ c - char. spl.
↳ b - block.spl.

- Eg:-
- \$ find . -type f | more ↴ (Search in Prod)
 - \$ find .. -type f ↴ (Search in Parent dir.)
 - \$ find / -type f ↴ (Search in Root dir.)
 - \$ find ~ -type f ↴ (Search in home dir.)
 - \$ find .. -type f -name "Empol" ↴ (Search by name)
 - \$ find . -type d | more ↴ (Search directory)
 - \$ find . -type f -name ".*" ↴ (Searching for hidden file)
 - \$ find /dev -type c ↴ (for Char. Special file).
 - \$ find -type f -Empty ↴ (displays all Empty files).
 - \$ find -type f -Empty -name ".*" ↴ (displays all empty hidden files).
 - \$ find . -type f -size +500 ↴ (Displays files having More than 50 characters). 19

\$ find . -type f -size 50c ↳ (displays exactly 50char files).
less than 50char means -50c.

\$ find . -type d -size 4K ↳ (displays the directories having size 4K).

Default size of a directory at the time of creation is 4K.

\$ find . -type f -perm 644 ↳ (Search files having 644 Permissions).

\$ find . -type d -perm 777 ↳ (Search directory having 777 Permission).

\$ find . -inum 164131 ↳ (Search files using Inode no.).

\$ find . -type f -links 2 ↳ (Search files having 2 links).

\$ find . -type f -links +2 ↳ (More than 2 links).

\$ find . -type d -links 2 ↳ (directories having 2 links).

\$ find . -type d -links -6 -links +2 ↳

\$ find . -type f -atime 5 ↳ (range)
(Searches files that are accessed in last 5 days).
by default it searches in days.

\$ find . -type d -mtime 5 ↳
(Search directories that are lastly modified.)

atmin → accessed in last min.

mmin → Modified in last min.

\$ find . -type f -maxdepth 2 ↳ (displays upto 2levels)

\$ find . -type f -mindepth 2 ↳ (displays from 2levels to last level).

```
$ find . -type f -empty -exec rm {} \; <
```

[It searches for all empty files and stores in buffer. Then the empty files are removed from buffer..]

exec. is used to execute a command.

```
$ find t1 -exec touch {} fi \;
```

(Create empty file fi in all the director of t1).

-OK is also used to execute a command but it will ask for Confirmation.

```
$ find . -type f -empty -ok rm {} \; <
```

Filters :- Extracting required data from the bulk of data.

- filters are always temporary. It doesn't save, just displays output.
- To Make Permanent we can use Redirections.

There are 2 types of filters.

Simple filters

more wc

less tee

cut head

sort tail

paste

tr

Advanced filters

grep

sed

awk

Simple filters:

more:- It is used to display output, Page wise.

- We use Pipe Symbol ' | '.

- Eg: Cat Emplist | more <

(or)

more Emplist <

When we Press Enter it moves Record by Record.

When we Press Space it Moves Page by Page.

b it Moves backward.

f it Moves forward.

If you give `$more <filename>`. If you Press V it will move to vi editor. to modify.

Press q it will quit from Command.

less:- It will work only in Linux.

- It ~~work~~ is used to display output Page Wise.

Eg: `Cat Emp | less <`

(or)

`less Emp <`

Same options as 'more'.

Cut:- We Can Cut characters and fields.

Eg: `Cut -c1 EmpList <`

it will Cut first character of each and Every row of your file.

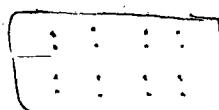
`$ Cut -c1,5 EmpList <`

it will Cut first and 5th character

`$ Cut -d ":" -f1 EmpList <`

"-d" is used to Mention the field Separator/delimiter.

If in a file the fields will be Separated by a Colon (:). Then to cut a Particular field. Here we are cutting first field which is Separated by Colon.



- We can use only one delimiter at a time.
- We can't give `::` also. for that we need to squeeze first.

Eg: Cut -d ":" -f1,5 Emplist ↴

Will Cut 1st & 5th fields.

Cut -d ":" -f1-5 Emplist ↴

Note:

In /etc/passwd we have 7 fields.

Username : Password : ^{User}Uid : ^{Group}Gid : Comments : HomeDir. : Default Shell
(X) ^{Where works}

indicates
it's in shadow
file.

↳ Cat /etc/passwd | Cut -d ":" -f1,3,6 ↴

Sort:

Note: in Vi editor Press Ctrl+v we will goto visual block. Then we can select and delete fields which ever we want.

\$ Sort -t ":" -K1 Emplist ↴

-t → Mention delimiter

-K → Mention fields.

- Will sort based on 1st character from ascending to descend.

\$ Sort -rt ":" -K1 Emplist ↴

- rt → reverse sort.

tr :- Character wise replacement.

Eg: \$tr "[a-z]" "[A-Z]" < Emplist ↴

Converts Lower Case to Uppercase.

\$tr "kol" "KOL" < Emp ↴

tr -s : is used for Squeezing.

Eg: \$tr -s ":" < Emplist ↴

tr -d : is used to delete.

\$tr -d ":" < Emplist ↴

\$ls -l | tr -s " " | cut -d " " -f3 ↴

longlist squeezing cutting.

\$ls -l | tr -s " " | cut -d " " -f6 | sort -M ↴

Sorting month field.

Paste :- Paste is used to get both files data.

Eg: Paste f₁ f₂ ↴

Eg: Paste -d "*" f₁ f₂ ↴

it will separates two file contents with *.

- To Paste Serially

Eg: Paste -s f₁ f₂ ↴

WC : (Word Count)

\$ wc Emplist <

143 176 3718
 Lines Words Char. Emplist
 filename

\$ wc < Emplist

143 176 3718

\$ wc -l → displays no: of lines

\$ wc -w → displays words

\$ wc -c → displays char.

tee :- is used to Create files.

Ex: \$ tee f1 <

Rapidshire
Rapidshire

Raj
Raj

\$ cat f1

Rapidshire
Raj

Can Create Multifiles With same data.

Ex: \$ tee f1 f2 f3

ctrl+d

Append data to Multiple files

Ex: \$ tee -a f1 f2 f3

Ex: \$ cat f1 | tee -a f1 f2 f3 <

head :- displays first 10 records

e.g. \$ head file

tail :- displays last 10 records

e.g. \$ tail file

e.g. \$ head -30 file | tail +25

'+' doesn't work in head

We can use '+' & '-' in tail

\$ head -30 file | tail -6

\$ head -30 file | tail -1

(for Particular Record).

ADVANCED FILTERS :-

* grep : (Globally Search for Regular expression & Print)

Syntax :-

\$ grep <options> "<Expr>" <filename>

-w	-H
-i	"[:upper:]"
-l	"[:lower:]"
-v	"[:digit:]"
-x	"[:Punct:]"
-s	"[:Space:]"
-e	"[:cntrl:]"
-F	"[:alpha:]"
-L	"[:alnum:]"

-w : Search for exact word and display the entire row.

Note: grep } only Row wise operations is Possible.
sed }

awk → both rows & field operations is Possible.

-i : ignores the case.

-n : wherever the expr. matches, it will print the line
and line no;

-c : Counts the no: of times the expr. is repeated in the
file.

-v : ignores the expression. ignore the things which we search
and print rest of the lines.

-r : Searches in a recursive manner. i.e., it searches in
Sublevels also.

only by using '-r' we can search in directory.

-s : to squeeze the errors.

-e : Searching for multiple expression. (for each expression we
have to give -e)

-E : Searching for multiple expression. (single time enough).

-l : If we don't know the filename where the expr. is searched.
So it will display the filename. (displays filename of expr).

-h : (same as -W).

-H : It will search in recursively. The output will be

<filename>: -----

<filename>: -----

- "[`:upper:`]": It only Search for Uppercase Expr.
 - "[`:lower:`]": It only Search for LowerCase Expr.
 - "[`:digit:`]": It only Search for number.
 - "[`:Punct:`]": Search for Special Char [`* # @`] and Prints that you
 - "[`:alpha:`]": Prints only alphabeticals.
 - "[`:alnum:`]": Prints both alphabeticals & numericals.

Sed : (Stream Editor)

Syntax:

\$sed <options> '<Expr|Record no.|actionpart>' <filename>

- n (suppressing duplicates)
- e (Search for multiple expr)

↓

- Printing (P)
- Deletion (d)
- Substitution (s)

awk : (Aho)

Weinberger

Kerningham)

Syntax:

`$ awk <options> '<|Expr|{Print}>' <filename>`

Action

-f → (field Separator)

-f- (To invoke script)

- By default it will accept spaces (any no.) as field separator. 24

Examples: (grep)

\$ grep -w "hyd" Emplist ↴

\$ grep -i "dba" Emplist ↴

\$ grep -n "dba" Emplist ↴
-ni

\$ grep -c "dba" Emplist ↴
-ci

\$ grep -v "dba" Emplist ↴
-vi

\$ grep -r "dba" * ↴

\$ grep -rs "dba" * ↴

\$ grep -e "hyd" -e "dba" Emplist ↴

\$ grep -E "hyd|dba|designer" Emplist ↴

\$ grep -l "dba" * | more ↴

↳ displays only filenames.

\$ grep -h "dba" Emplist ↴

\$ grep -H "dba" Emplist ↴

\$ grep "[[:upper:]]" Emplist ↴

lower

digit

Punct

alpha

alnum

\$ last ↴

- displays about login, logout, users & booting info.

O/P: Username terminal ip address of time
thin client

\$ tty ↴

- displays terminal no.

\$ last | grep -w "down" ↴

- displays all shutdown activities.

Examples: (Sed)

\$ sed 'IP' Emplist ↴

\$ sed -n 'IP' Emplist ↴

\$ sed -ne 'IP' -e 'lop' Emplist ↴

\$ sed -ne '/dbalP' -e '/designerP' g! ↴

\$ sed 'Id' g! ↴

- delete 1st record

- for deletion no need to use -n

\$ sed '1,sd' g! ↴

\$ sed -e '/hyd/d' -e '/dbal/d' -e '/designer/d' g! ↴

\$ sed +n 's/hyd/india/g' g! ↴

it will search for hyd & substitute by india.
in the first occurrence only.

\$ sed 's/hyd/india/g' g! more ↴

globally.

\$ sed -n 's|hyd|india|2p' |g| ↴

Second occurrence.

\$ sed -n '1s|hyd|india|P' |g| ↴

only first record.

Uniq → by using this filter it will avoid multiple lines. It will print single line.

Examples :- (awk)

\$ awk -F ":" '{Print}' Emplist ↴

\$ awk -F ":" '{Print \$0}' Emplist ↴

↳ Indicates field

\$ awk -F ":" '{Print \$1}' Emplist ↴

Print & Print \$0 is same

'{Print \$1,\$3}'

Range - does not work.

\$ awk -F ":" '|dba|{Print}' Emplist ↴

\$ awk -F ":" '|dba|, |designer| {Print}' Emp ↴

,

- indicates range.

\$ awk -F ":" 'NR==1 {Print}' Emplist ↴

'NR==1, NR==5 {Print}'

\$ awk -F ":" 'NR==1, NR==5 {Print \$1,\$3}' Emp ↴

\$ awk -F ":" '\$4 > 20000 {Print \$1}' Emplist ↴

- Prints field 1 whose field 4 is greater than 20,000

\$ awk -f " '\$4 >= 10000, \$4 <= 15000 {Print \$1}' Emp

\$ awk -f " {Print NR, \$0, NF}" Emp.txt
Line no. no. of fields

\$ awk -f " NR == 5, NR == 9 '{Print NR, \$0, NF}' Emp

\$ awk -f " NR < 5 {Print}" Emp.txt

\$ awk -f " NR == 2 || NR == 5 {Print}" Emp

only 2nd & 5th row will print.

'NR == 2 || NR == 5 {Print \$1, \$4}'

\$ awk -f " '\$3 > 8000 {Print \$1, \$3}' /etc/passwd

Process :- Any Program under execution is called Process.

- Default shell process will run when a user login & until he/she logs out.

\$ ps ↳ displays Current Process running in your System.

Op: Pid terminal Datetime bash

\$ ps <options>

-a

-e

-f

-c

-x

Foreground Process :-

- we can perform only 1 task

Background Process:-

- we can perform 'n' no. of tasks.

e.g. \$ cp file1 dir1 & ↵

o/p: [Jobid] [pid]

& - is used to send the process to background.

\$ Jobs ↵ :- to know the status of the background process.

\$ fg <Jobid> :- The background process will come to foreground.

\$ bg <Jobid> :- again go back to background.

\$ Ctrl + Z :- Stop Process. You will get Jobid.

\$ Sleep 100 :- You will not get '\$ Prompt' for 100 sec.
100 minutes.

Killing Process :

\$ Kill -9 <Process id> ↵

-9 indicates sureshot killing.
only using kill, we don't know
whether it's killed or not.

\$ top ↵ gives all the details of Process.

COMMUNICATIONS :-

Offline → Mail.

Online → Write

TALK }
Ytalk } 2-way

Write → one way communication.

\$ write <userid>

\$ mesg n → to disable your Comm. tool.

\$ who -T → displays who all are online.

+ → enabled.
- → disabled.

+ → We can send message.

- → disabled Comm. tool so we cannot send msg.

\$ who -TH → displays Human Readable format.

\$ mesg y → Comm. Tool will be enabled.

| Var | Spool | Mail → The Mails will be stored here in this Path.

OFFLINE :

\$ mail <userid>

Sub:

ctrl+d ↵

cc:

ctrl+d ↵

Message will go through terminal
after 60sec. by default.

Check Mail

`$Mail ↴`

?d ↴ Delete the Message.

?l ↴ List from Mail.

All messages are stored in Mailbox.

To read the Mailbox file

`$Cat Mailbox ↴`

To delete particular message.

`$ <no> d ↴`

`$ talk <uid> ↴` Reply at same time.

Wall → Admin Command.

Error: You must be super user.

Wall ↴

Normal users under server will get a message at a time by using Wall.

Input Re-direction: (<)

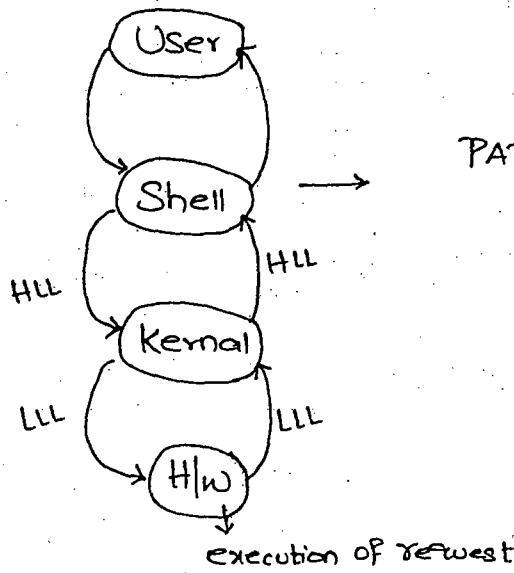
Any operation performed on a file.

Eg.: Cat < f1 ↴

2 > Error redirection
→ field descriptor

1 > Output redirection
→ field description.

Shell :- is a Command Interpreter | Command Checker.



$\text{PATH} = \text{:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin}$

`$ which <Command Name>`

- It gives entire Path of Command. Where it is

located in the ~~directories~~ directories.

Types of Shells :-

Default OS

bash (bourne again shell) — linux

sh (bourne shell) — SunSolaris & HP-UX

ksh (Korn shell) — AIX - IBM

csh (C shell) — IRIX

tcsh (Turbo C shell)

zsh (Z shell)

- Linux Supports all the shells. Default it is bash. but it supports all.

- Solaris supports only bash. default is bourne shell. it supports bash shell only

- Hp-ux doesn't support any other shell. Default is Bourne Shell.
- Aix-IBM doesn't support any other shell. Default is Korn Shell.
- IRIX - doesn't support any other shell. Default is Cshell.

Location of all Shells:-

\$ cat /etc/shells ↴

- display all the shells.

(or)

\$ ls /bin/*sh ↴

(or)

\$ chsh -l ↴

* How to Change Shell Permanently ?

\$ chsh ↴

Password: ↴

/bin/bash :/bin/sh ↴

- shell changed.

to check whether changed (or) not.

Just logout & login

Temporary

Eg: bash -bourne
\$sh ↴

bourne - korn
\$ksh ↴

then

\$echo \$SHELL

Printing Purpose.

- To View in Which Shell We Are In.

Variable :- Which Stores Some Values Is Called Variable

\$a = 10

\$echo \$a ↓ - To Print The value of a.

\$a = 10

\$b = 5

\$c = `expr \$a + \$b`

→ Shell Command.

\$c = 'ls'

\$echo \$c

\$ echo \$c ↓

15

- Subtraction

* Multiplication (only * is a Wild card char.)
/ division.

\$env → displays System & Environmental Variables.

\$set → displays System & User defined Variables.

Positional Parameters :-

Passing an Argument to a Command

is Called Positional Parameters.

e.g.: Cat > f1 ↓
→ Positional Parameter

\$set Jan feb Mar apr May jun jul aug Sep Oct Nov

Dec.

\$ echo \$1 ↓

Jan

\$ echo \$10 ↴

100

So for this we have to use

\$ echo \${10} ↴

\$ uname ↴ - displays OS.

\$ uname -a : displays details.

OS Hostname Version Date Modelname Processor type.

\$ uname -m : displays Model type.

\$ uname -P : displays Processor type.

\$ uname -r : displays Release Version.

\$ uname -s : displays OS

\$ who ↴ : displays Users Logged in

\$ who -i : displays idle time also.

\$ who -H : displays in Human Understandable format.
(displays col. names).

\$ who -T : Who are all available to communicate &
Non Communication.

`$ mesg -n`: to display Comm. tool. + → -

`finger`: displays more details about User.

`$ finger`

to change finger info.

`$ chfn`

↳ This changed info. is stored in etc directory
in the Comment field.

`$ finger -l ; more`

↳ : long list format of each & every user

* Last: Entire info. about users who are logged in &
who are logged out.

- What time they logged in & logged out.

Reboot details etc..

`$ last`

Last Comm:

`$ lastComm`

displays what is the last command that a user have
performed.

Shell Continues...

To See System Variables & user defined. System Variables.

\$ Set

System Variables are in Caps.

to See Particular System Variable.

\$ echo \$<System Variable>

\$ echo \$HOME — displays home dir. Path.

\$ echo \$HISTORY — by default it Stores Previously used
1000 Cmds.

in Aix - 126

HP-UX - 118

Linux - 1000

Solaris - 500

\$ echo \$a : to see value of 'a'.

e.g. vi fi

echo "Enter the first no."

read a

echo "Enter the first no."

read b

echo "The addition is \$a + \$b"

! run!

\$ sh fi to execute.

SHELL

SCRIPTING

Lecturer: Sunder

9985778710
01/08/14

JIMP II BASICS

FAP

LINKS

PROCESSES

FILTERS

FIND

Scripting Languages

- Unix Shell Scripting
- Pearl Scripting (Practical extraction & reporting languages).
- Python
- TCL (Tool Command Language).

Frequently Used Shells in Real Time are

- Kom Shell
- Bash Shell

Runlevels (Solaris)

init 0 → Shutdown | Firmware (Interface to interact with H/w).

init 1 → Single User Mode.

init 2 → Multiuser Mode.

init 3 → Multiuser Mode with NFS.

init 5 → Shutdown

init 6 → Reboot

/etc/inco.d/* → init 0

/etc/incl.d/* → init 1

/etc/incl2.d/* → init 2

Scripting Advantages:

- Poor GUI

- Not Accurate (because no datatypes)

* In Scripting Everything will be stored as Strings.

Simple filters:

- More

Eg.: More <filename> ↴

to modify there Press V it will go to vi ed

- less (only in linux)

- Head

- Tail

Tail -f <filename>

: If any updates are going on in a file. it will display that.

- WC (WordCount)

wc -L fi ↴ : displays length of the longest line.

-\$ ls -1 | wc -l : displays total no: of files.

display in
seq. way
line by line

- tee (same data). Permanent filter

\$tee a1 a2 a3

\$tee -a a1 a2 a3 (Appending data)

abcd

efgh

ijkl

- Cut (to get data field wise)

Cut -d (delimiter) Cut -d " " -f1,2 -f6 ↴

-c (charactor) Cut -c 6 -fc 4

- Sort

\$ sort -t " " -k3 file1

\$ sort -nt " " -k4 file1
~~~~~  
numeric  
Sorting

\$ sort -Mt " " -k1 file1  
~~~~~  
month
Sorting

- tr (translate)

\$ tr "a" "A" < file1 >

- Converts all a's to A in file1.

\$ tr "[a-z]" "[A-Z]" < file1 >

\$ tr -s " " < file2 >

Squeezing

- Paste (display contents of a file adjacent to each other)

Paste file1 file2 >

5/9/09

Advanced filters:

grep : (globally search for regular expression & print) } Row level

sed : (stream editor) } Operations

awk : (Aho, Weinberg, Kernighan) → Row & Column level
Operations.

Grep:

Ex: grep "tester" file

grep -w "with" file : displays Exact Match.

grep -n "Chennai" file : displays no. to lines

grep -i (Ignore the Case)

grep -c : (Count no. of times the word exists)

grep "Aa" : displays lines starting with 'a'.

grep "o\$" : displays lines ending with o.

\$ ls -l | grep "Ad" : displays directory files.

-The files which cannot store data is character Special files.

Ex: Keyboard, Mouse, Monitor etc..

The files which can store data is block Special files.

Ex: Harddisk, Cd, floppy etc..

\$ grep -e "tester" -e "analyst" file

-Searching Multiple Patterns.

\$ grep -E "tester|analyst" file

\$ egrep "tester|analyst" file

\$ grep "Delhi" *

- Can Search in all files in Pwd

- To Make the Opt Permanent We can use Redirections.

Sed: (Stream editor) :

1. Printing the text (P)
2. Deleting the text (d)
3. Substituting the text (s)
4. Inserting (i)
5. Appending (a)
6. Changing (c)

1. Printing (P) ::

Syntax:

```
$ Sed '<actions>' <filenames>
```

<actions> Can be no; | String Patterns

Eg: \$ Sed '5q' file1 ↴
 ^quit

\$ Sed '3P' file1 ↴

- it will Print all lines but 3rd line twice.

\$ Sed -n '3P' file1 ↴ : it will Print only 3rd line & only once.

\$ Sed -n '3,\$P' file1 ↴ : Print from 3rd line to last line
of a file.

\$ Sed -n '10,\$!P' file1 ↴ : 1st line to 9th line.

\$ Sed -n '1~2P' file1 ↴ : Print only odd lines.

2. Deleting (d):

\$ sed '1,5d' file1 : deletes 1-to 5th line.

\$ sed '1~2d' file1 : deletes odd lines.

\$ sed -ne '3p' -e '6P' file1 : Prints only 3rd& 6th line.

\$ sed -n '/dbalp' file1 : Searches dba & Prints that row
in order to give string we have
to give in 2 forward slashes.

\$ Sed -ne '/dbalp' -e '/ualp' file1 : Multiple Patterns.

3. Substituting the text (s):

\$ sed 's|analyst|ANALYST|' file1

- displays ANALYST in place of analyst
only in first occurrence.

- to display everywhere.

\$ sed 's|analyst|ANALYST|g' file1 : globally.

- to Substitute only in 2nd Occurance use 2
in place of g

- to Substitute in Particular line only then.

\$ sed '1,4s|analyst|ANALYST|' file1 : Substitutes from 1st to
4th lines.

\$ sed -e '1s|analyst|ANALYST|' file1

- only in a Particular line

4. Inserting (i) :-

\$ Sed '31

>Hi how are you!

Y - - - - -

> 'file1' ↵

In file1 in 3rd line this text will be inserted and 3rd will become 4th.

S. Appendix (a):

\$ Sed 'hal

> Have a nice day

> file ↵

- Chang the 3rd Line.

AnsK :-

- It can be used as a filter.
 - It can be used as a Programming Language Tool.
 - accept all Syntaxes of 'C'.

Syntax:-

\$ awk <options> ' <action>' filename ↓

Frank 'Spinty' file ↩

\$ awk -F ":" '{Print \$1, \$3}' file

`$ awk -F ":" 'NR==3 {Print $1, $4}' file1`

→ displays 1st & 4th field of 3rd row.

`$ awk -F ":" 'NR==3, NR==5 {Print $0}' file1`

→ displays 3rd & 5th row.

`$ awk -F ":" 'NR==3 || NR==5 {Print $1, $4}' file1`

→ displays 3rd & 5th row.

`$ ls -l | awk -F " " '{Print $1, $9}'`

`$ awk -F ":" '$4>=15000 {Print $0}' file1`

`$ awk -F ":" '$4>=15000 {Print NR, $0}' file1`

→ row no: will also be displayed.

To display latest file `$ ls -lrt`

`$ awk -F ":" 'NR==2, NR==5 {Print $0, NF}' file1`

→ displays no: of fields each row have.

`$ awk -F ":" '{Print $NF}' file1`

→ gives last field.

`'{Print $(NF-1)}'`

→ gives last but one field.

`$ awk 'length>10' file1`

→ displays lines greater than 10 char.

Search Using Strings:

```
$ awk -f ":" '| developer | { Print $1, $2, $4 }' file
```

— Search for developer and prints 1,2 & 4th field.

```
$ awk -F ":" '| developer ||| tester | { Print $1 }' file
```

```
$ awk -f ":" '$4 >= 15000 && $4 <= 20000 { Print }' file
```

Task: How to Search all .100 char. files & Remove them.

vi act1

```
$4 < 15000 { Print $1, $2, $4 }
```

```
$ awk -f ":" -f act1 file1
```

```
$ awk '{ FS = ":" } { Print $1, $2 }' file1
```

Can change field Separator in O/P.

```
$ awk '{ FS = ":" ; OFS = "#" } { Print $1, $4 }' file1
```

| output
| field
| separator
|

Another way of Searching String

```
$ awk -f ":" '$2 == "Chennai" { Print $0 }' file1
```

— We can give User defined Statements.

BEGIN }
END } Keywords.

Veritas Operations :

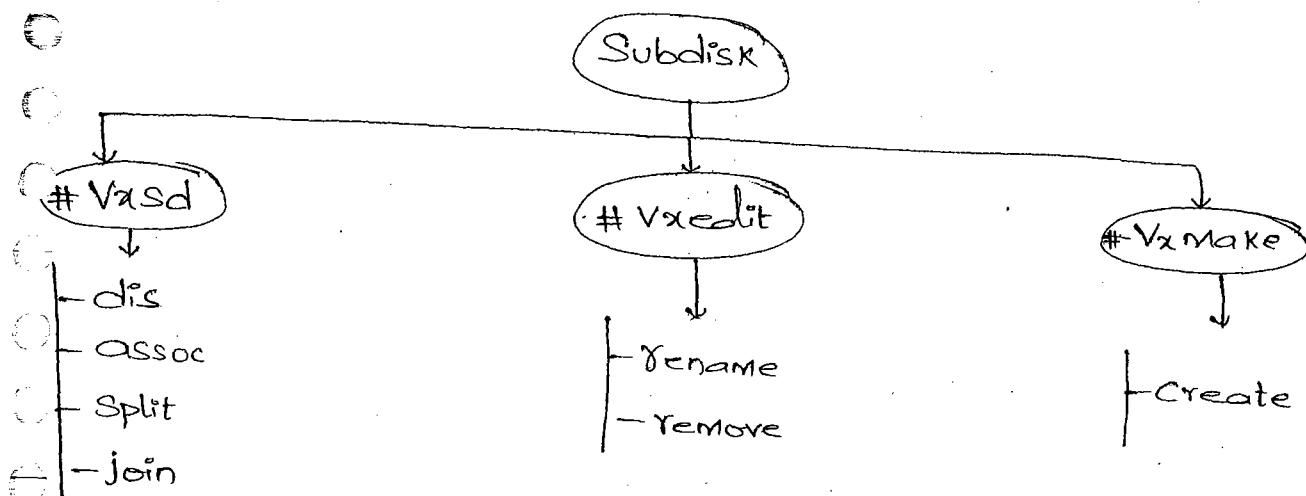
We Can Perform 3 different types of Operations

1. Subdisk level

2. Plex Level

3. Volume Level

1. Subdisk level operations:-



All the Subdisk level operations are not Suggestable because We Can't give data guarantee.

#Vxmake: By using this We Can Create Subdisks.

Eg: Vxmake -g Oracle Sd Sdi do,0,100m ↴

#Vxedit: By using Vxedit we can Rename & Remove Subdisk.

Eg: #Vxedit -g Oracle Rename <old.name> <new.name> ↴

5/11/01

Vxdg -q <dgname> free ↴

: list disk group offset value

(or)

Vxdg free ↴ : All available diskgroups free space

Vxinfo -q <dgname> ↴ : Complete info. about diskgroup.

Ps -ef | grep VxConfigd

VxConfigd -K ↴ : Restart the Configuration daemon.

Vxdctl Mode ↴

o/p: disable

o/p: booted

Vxdctl enable ↴

Vxdctl disable ↴

Vxdctl enable ↴

How Can we make default disk group:

Vxdctl defaultdg <dgname> ↴

Vxdg destroy oradg

Vxdisk -s list | grep cti do ↴

dg.id → 11913447.wstsunt

Vxdg import 11913447.wstsunt ↴

Vxdisk list ↴

Start all the volumes & Mount it

If we destroy the dg
but still we can get the
data. because dg.id is
there.

by importing it.

Eg:- Vi act2

BEGIN {Print "The Employee details are:"}

NR==2, NR==5 {Print \$0}

END {Print "End of Employee details"}

\$awk -f ":" -f act2 file2 <

Types of Shell:

\$ cat /etc/shells — in Linux

\$ ls /bin/*sh — Other Os.

OS	Type of Shell
Linux	bash

Solaris (Or)	Bourne Shell (sh)
HP-UX	

IBM-AIX	ksh
---------	-----

IRIX	csh
------	-----

Changing Of Shell:

(i) Changing the Shell Temporarily

(ii) Permanently.

\$ ps -l — know the kind of shell using.

To change to Kom Shell

\$ ksh <

II) Permanently :-

\$ chsh

Passwoord:

\$ /bin/ksh <

Shell Successfully Changed.

/etc/passwd - Contains User info. (total 7 fields).

/etc/shadow - Encrypted Passwords of Users.

Startup (or) Initialization files:-

Linux (bash)	• .bash_profile, .bashrc, .bash_history, .bash_logout		
Solaris (Bourne) (sh)	↓	• Profile	• .exrc • .cshrc
Aix (ksh)	↓	• Profile	• .Kshrc
IRIX (csh)	↓	• login	• .cshrc, .logout

Whenever a user is logged in these files are seen in home directory.

/etc/skel - all the startup files are copied from this path when a user is logged in.

<u>/etc/profile</u> (all Os)	- When a User is logged in the environment which he gets is from this Path. (or)
<u>/etc/bashrc</u> (only in linux Os)	(System wide Settings are Present in this file)

Aliases: Aliases are Set in Startup files i.e., bash_profile.

Eg: \$alias c='clear'

\$unalias c - will remove alias for that session.

\$alias - will display all aliases

Ways to execute a Script:-

\$Sh <filename> → : bourne Shell & all

\$. <filename> → : Present Working Shell

\$ Ksh <filename> → : Korn Shell

\$ bash <filename> → : bash

\$ Csh <filename> → : Cshell

\$.!<filename> → : If we have execute Permission.

Variables:

- User defined Variable
- System defined Variable
(or)

Environmental Variables.

* User defined Variable:

a = "Sun"

- There should not be any Space before & after assignment (=) operator.

```
$ echo $a } Print $a  
$ echo $n } Print $n
```

- Should be in Small letters only.

* System defined Variables:

SHELL

HOME

PWD

PS1

PS2

PS3

PS4

HTSTSIZE

PATH

LOGNAME

\$env ↴ : all Environmental Variables.

(or)

\$Set ↴ - displays all Environmental Variables as well as User defined Variables.

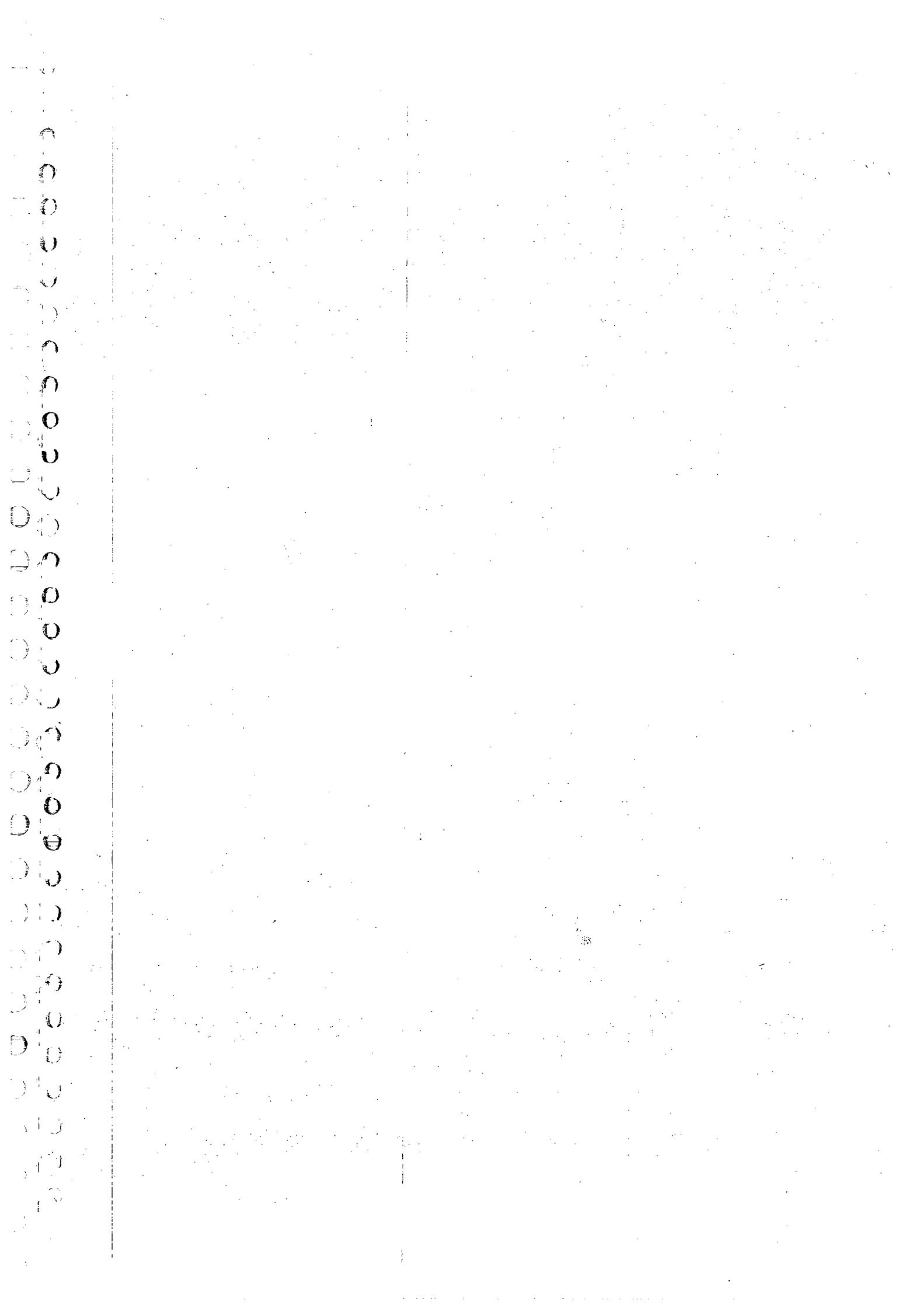
- They will be in Capital letters.

\$echo \$SHELL

- displays the Shell User using.

\$echo \$HOME

- displays Present Working Directory.



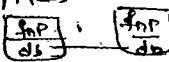
5/10/09

UNIX ADMIN

dd > <filename>

links.

files



dir.



\$ ls -ll <targed file>

& ;

using these symbols
in this the commands you can
execute multiple cmds.

1. Vi editor

2. find

3. grep

4. FAP

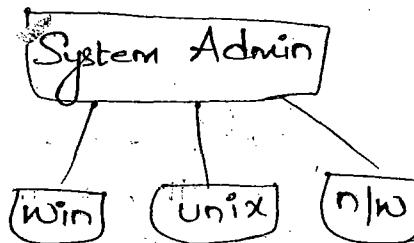
5. CP, RM, MV

6. Links

It will display the Perm. of Softlink file.

With out giving execute Permission also we can execute using
execute using ./<filename>. You need execute Permission.

Sh <filename>. To



Sys. Admin is a Person who is Employed to operate a Computer / Server / Network.

Server: A System which is Providing Services to one

(or) More System.

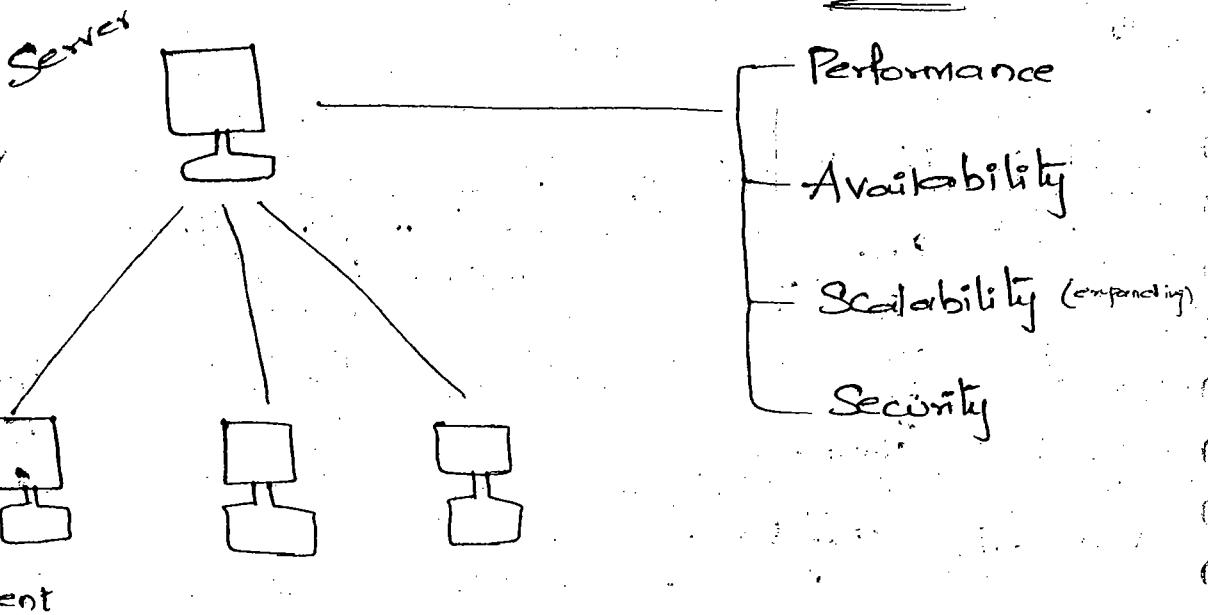
Application Server

Web Server

file Server

etc.,

Client: A System which is accessing the services from Server.



Performance

Availability

Scalability (Pending)

Security

- having good no. of resources improves Performance.

- 24x7 Availability.

Clustering: We have 2 systems. One server will provide services to clients. If that server fails, then without downtime clients will receive (or) avail services from other one. This is clustering.

- Scalability: We can expand, if there is increase in no. of ~~users~~ users then server should be in the position to expand (or) provide facility to others also.

Security

- Hardware level (based on physical locks)
- OS level. (Userids, groupids) etc,

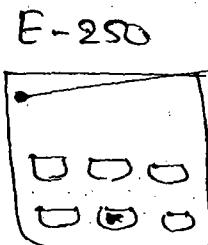
Hardware level.

Hot Swappable

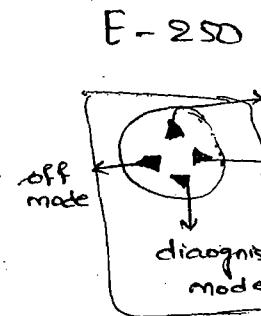
Hot Plugable.

If a Server is having 6 harddisks and one hard disk is crashed we can remove that harddisk and place new this is Hot Swappable.

We need to open the physical lock first and then we can change (or) plug in new one. Hot pluggable.



→ Physical lock.



on mode
off mode
non-secure (dial-in) mode.
diagnostic mode
we can access CPU memory without sending warning.

Difference between Workstation & Server:

Workstation

Ram → 4-8GB

PCI → 2-4

CPU's → Mod. 2

HD → 4 disks

I/O → T.D.E, S.C.S.I

Integrated device electronics

Small computer System Interface

Server

9 terabytes

192

128

75 internal disks

S.C.S.I, F.C

fibre channel.

Tasks Of Administrator:

- * Installation
- * Upgradation
- * Alc Mgt.
- * Package & Patch
- * Backups
- * RAID
- * Monitoring.

Note:

for each & every command it have binary executable files internally.

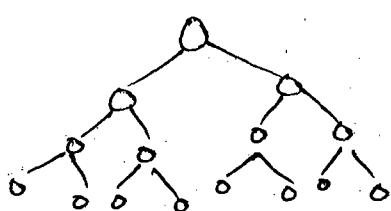
To Kill a Process we use Kill Command.

Kill -9 → is Sureshot Killing

Kill -8 → is abrupt killing.

For example
When we want to copy a file of size 5gb to another directory it takes time.

As it is a big process it will divide the job into smaller jobs



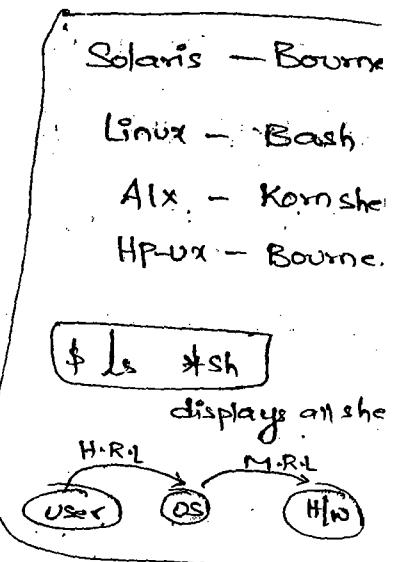
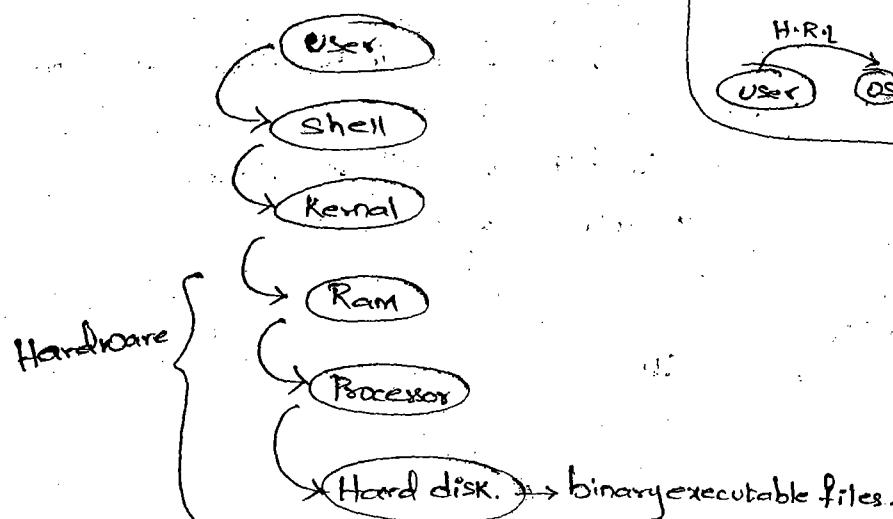
and now if we try to kill this process ~~the~~ using Kill -8 there is a chance that ~~the~~ any one child process may not be killed. that process is called Zombie Process / Orphan.

6/10/u

Hardware Architecture:

- Based on MotherBoard We can tell what is a server & Workstation (PC).

CPU:-



We have 2 types of Processors

- * CISC (Complex instruction set code/Computing) based Processor
- * RISC (Reduced instruction set computing) based Processor.

- All X86 Processors Comes under CISC.

- IBM Processor, Sparc, HP Comes under RISC.

& Performance is Main difference b/w both CISC & RISC.

- In CISC based Processors We can Install any OS. Such as Windows XP, Solaris, Linux etc,

- In Risc based Processors we have only Proprietary OS. that means in IBM Processor we can install only AIX. In Sparc Processors we can install only Solaris.

In HP Processors only HP-UX. No Other OS.

3. In

CISC - { 32 bit - 2^{32} instructions at a time
64 bit - 2^{64} at a time.
(Mainly used in Servers)

RISC - { 32
64

.. In X86 Processor Machine b/w have

Intel

8086

80186

80286

80386

80486

P-I

P-II

P-III

P-IV

Dual Corp

Core 2 Duo

Quartz Core.

} These all Processors we can use in PC. i.e., Workstations.

In Windows Servers Itanium I Processor
 " II Processor.

- In "Sparc" (Scalable Processor Architecture), Processor

We can install only Sol. os.

- In IBM we have "Power" (Performance optimization with enhanced Risk) Processor. Here we can install only AIX OS.

- In HP we have "PA-Risc" (Precision Arch. Reduced Instruction Set Code) Processor. Here we can install only HP-UX OS.

In "Sparc" we have

↓
UltraSparc - I

UltraSparc - II

" III

" IV

" IV +

} mid range servers

After releasing UltraSparc - IV +

Processors Sparc Processors

released "Cool Threaded" Processor

↓
T1 } Tseries.
T2

- Cool Threaded Processors are meant only for High end servers.

- Using UltraSparc Processors we can install only 1 Solaris OS.

- In Cool Threaded Processors we have logical domains. In each logical domain we can install diff. OS.

Win xp OS	linux	Solaris
--------------	-------	---------

l'domes.

Why we go for Cool Threaded Processors?

- Because each and every Ultrasparc Processor.

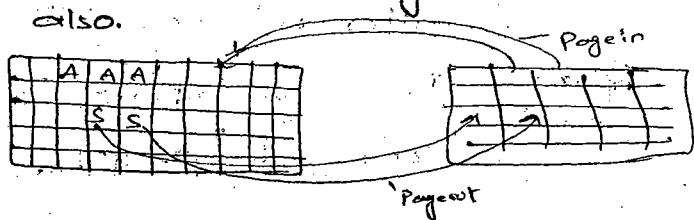
Each processor has its own memory & I/O

- We can call this entire system as Hardware Virtualization

In I-domes ~~so~~ each & every OS acts as different servers.

RAM :-

- Ram is made up of Registers. We can call these Registers as Pages also.



Dif. Type of Rams:

Sd Ram

DDRRam.

- We have 2 types of Slots for Ram.

(Single Inline Memory Module) SIMM

(Dual Inline Memory Module) DIMM

- Always Physical Ram should maintain Active Status only.

Otherwise it will hangup. not sleeping Processor

- If System Performance Comes down then we have to Create a virtual Ram on hard disk. When we Create a Virtual Ram all the sleeping Process in Physical Ram

Will Move to Virtual Ram. This is Called Page out. This whole thing we call as "Paging". After becoming the Sleeping Process to Active Process it will move from Virtual memory to Physical memory. This is called Page in.

In x86, if we have 2 512 RAMs in our system. If one RAM doesn't work then System won't turn on.

But

In Sparc Processors the System will turn on.

E.C.C (Error Code Correction).

Cache Memory:

Interface b/w CPU & Mainboard.

e.g. \$du -sh /dirl ↴

Let us suppose after 1½ min it displays 5GB and immediately if i execute same command again, it will display result immediately. as the last performed command is stored in Cache.

- Cache memory, Maintain the Previously accessed info.

We have 2 types of Cache memory.

L1 (we can see in Mainboard)
L2 (in Processor)

Hard Disk :-

It uses P/Io ~~controller~~ Technology

IIDE

PATA (Parallel)

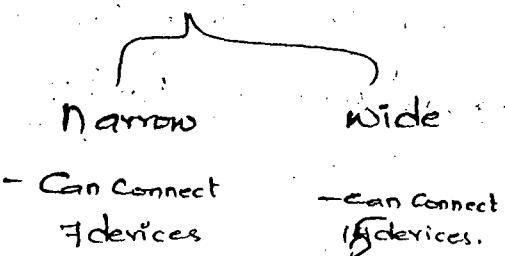
- multiple.

- Speed is low

SATA (Serial)

- one controller - one device
- Speed is high

In SCSI



PCI slots are used for installing VGA Cards and external Cards.

Types of Servers :-

- Based upon Performance & Scalability. They are using different types of servers.

* Entry level servers

* Mid Range

* High End

www.Sunsolve.com
docs.SUN.com

Device Naming Conventions:

Solaris OS we can install on

two platforms X86 & Sparc.
(Intel)

In X86:

$$\begin{matrix} C \# & d \# & s \# \\ \downarrow & \downarrow & \downarrow \\ \text{Controller} & \text{device} & \text{slice} \end{matrix}$$

S — Slices / Partition

In Sparc:

$$\begin{matrix} C \# & t \# & d \# & s \# \\ \downarrow & & & \\ \text{target} & & & \end{matrix}$$
Controller:- It is an interface b/w CPU and disk.

- It lays the communication b/w disk and CPU.
- We have different type of Controllers

(i) IDE

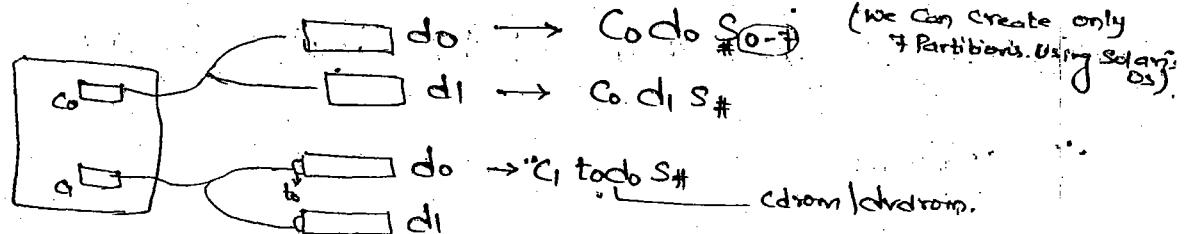
(ii) SCSI

(i) IDE Controller (Integrated Device Electronics):

we have 2 types

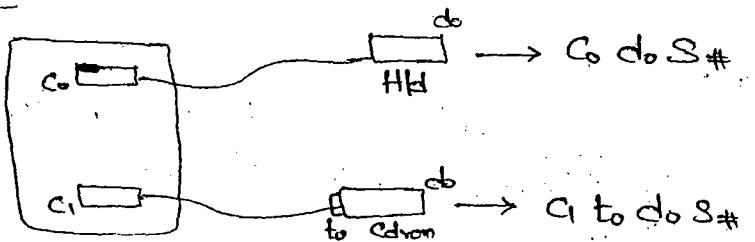
IDE

- PATA (Parallel Advanced Tech. Attachment)
- SATA (Serial Adv. Tech. Attachment).

PATA:(We can create only
7 partitions using Solaris
OS.)

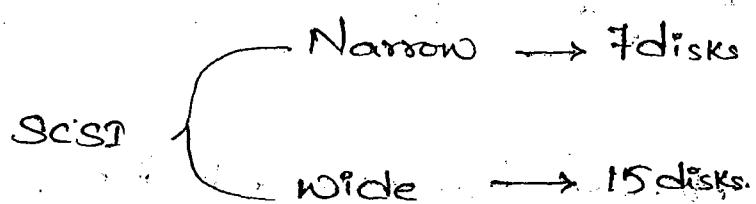
- If we want to connect more than 4 hard disks then we have to use SCSI Controller externally.
- IDE can find out target only for CD-ROM drives in X86 machine.

SATA:

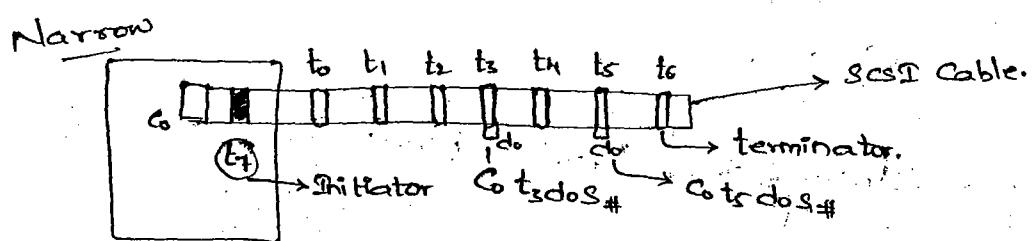


In SATA only one Controller for one device.
It is speed than PATA.

ii) SCSI (Small Computer System Interface):



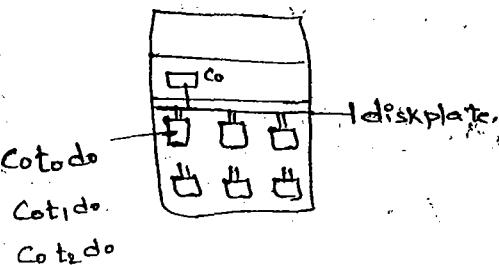
From Ultrasparc - 60 we have SCSI Controller. before that IDE Controller.



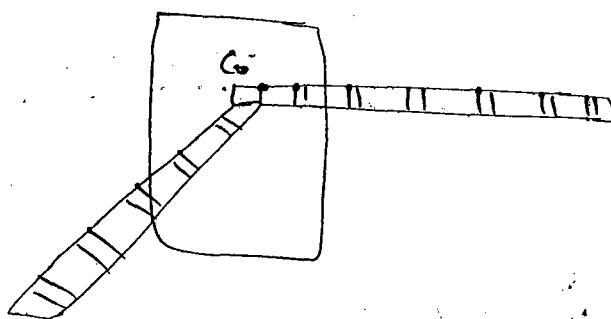
In each target we connect cdrom / hdisk ..

`#format -l` is used to display the no: of hard disks
`#cgadm -al`

Terminator is used to send back signal to controller.



wide SCSI



Target : (t#) Target no: Such as t_0, t_1, t_2, \dots Used to identify the hardware address of a disk, cdrom, tape.

Disk : - (d#) Disk name represents L.U.N (logical unit No:).

If we go with SAN (Storage Area Network) then there will be change in disk name.

Slice : (S#) - Smallest Partition on a Hard disk.
- At Max. we can Create 0 to 7 Slices.

$S_0 - S_7$

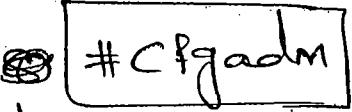
S_0
 S_1
 $\boxed{S_2}$ → Reserved (it contains all slice info.)
 S_3
 S_4
 S_5
 S_6
 S_7

Device Configuration:-

We can Configure device in 2 ways.

1. Command line Interface ~~(CLI)~~ (CUI)

2. Graphical User Interface (GUI)

 **#Cfgadm** is a command used to Configure and unConfigure.

In Solaris to refresh we have a Cmd: **#Xrefresh**

Admin tool → Sol. 9 } have GUI
SMC → Sol. 10 uses

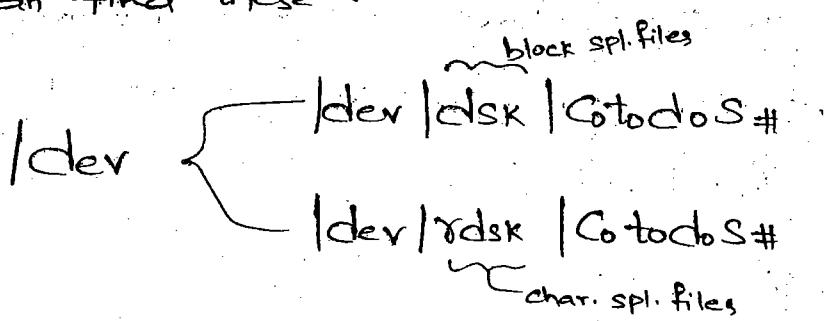
(Solaris Mgt Controller)

When we attach a New disk device (PenDrive, Printer, etc.)

Automatically 2 files are generated.

- Block Special File
- Character Special file.

We can find these files in /dev directory.



```
# cd /dev/dsk \
```

```
# ls -ll /etc/dso#
```

* Whatever files located under /dev/dsk and /dev/rdsk
are softlink files.

* When a disk is attached it will ^{new} _{update} resemble in 3 location

- 1) /dev
- 2) /devices
- 3) /etc/Path-to-inst

1) /dev : Responsible for maintaining all devices logical names.

The names under /dev/dsk & /dev/rdsk are logical names. e.g.: /etc/dso#

2) /devices : - Responsible for maintaining all physical names of devices.

- Physical names are for Kernel.
- Kernel can identify devices with the help of physical names.

3. /etc/Path-to-inst:

Responsible for maintain devices Physical Names as well as Instance Names (device module info.)

- Module is nothing but device drivers.
- If we insert any nic Card then we can find it here.

```
# Cat /etc/Path-to-inst | grep net ↴
```

- It maintains Physical names and Instance Name (ie, device driver name).

- At the time of boot up the Kernel will try to load the drivers from this Path. (/etc/Path-to-inst)

Kernel will load the drivers and update the physical name, so that we can identify that the device is connected.

- If we remove this /etc/Path-to-inst it will have a backup like /etc/Path-to-inst.old.

- If we remove these 2 files, we have to boot in ReConfigurable booting, then it will execute the following

Command OK > boot -r & it will be created.

Device Detection :- (detecting devices).

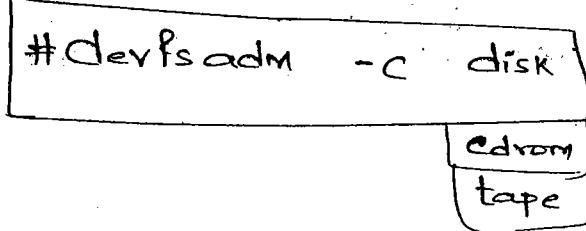
- Online detection (With out Rebooting we can detect) without downtime
- Offline detection (With Rebooting)
- Online detection:

```
# devfsadm
```

device filesystem admin.

After executing this Command it will detect all the newly attached device

```
# devfsadm -c disk
```



-c to specify the class of the device.

```
# Volcheck
```

to detect any removable devices.
(Pendrives).

After executing # devfsadm it, will not give any output.

Then execute

```
# format  
# Cfqadm -al  
# iostat -en
```

it will confirm whether there is any device detected or not

If we execute # devfsadm -c net ↴

```
# letc | Path-to-inst | grep net ↴
```



(con)

(tun)

s0g3

format

contd...

#devfs admin.

Hotpluggable: If any I/O operation is ^{not} running we can remove & place new disk.

Hotswappable: If any I/O operation is running we can replace disk.

Offline detection:-

We have to take the downtime.

If we want to reboot the machine after attaching devices then it will be dealing with the existing devices.

reboot -- -r

- reconfigurational reboot.

(or)

OK > boot -r

at bios level.

touch /deconfigure

then # reboot <

After rebooting we will not find deconfigure file.

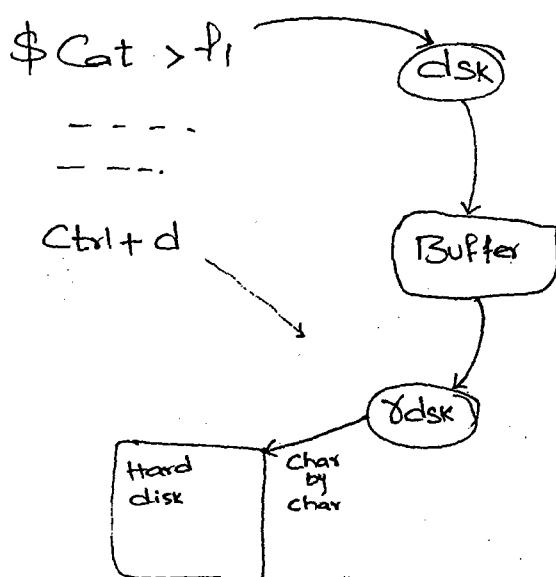
8/10/09

Block Spl. file (dsk)

Char. Spl. file (xdsk)

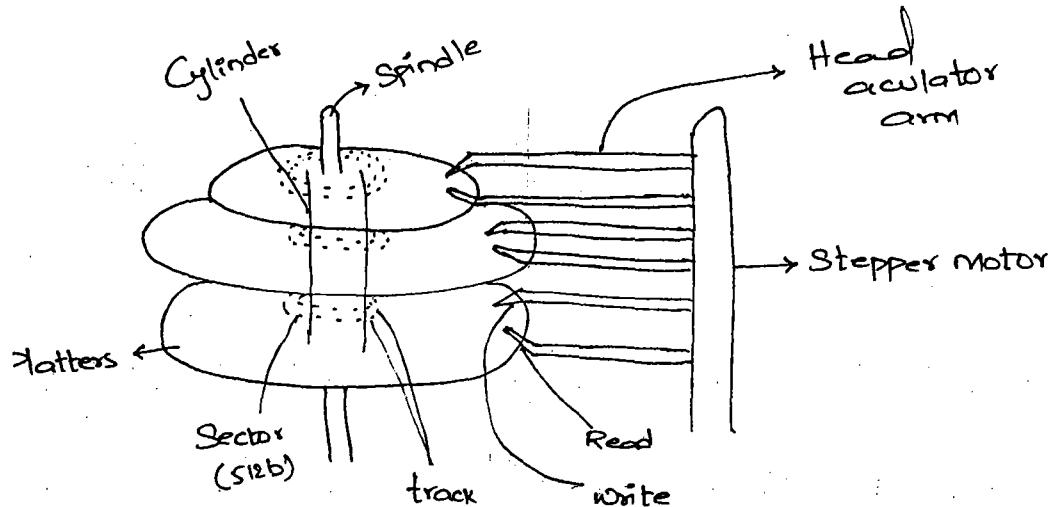
dsk is responsible for allocating datablocks to a file.

xdsk is responsible for writing data character by character in hard disk.



Hard Disk Layout :- Hard disk is a collection of Magnetic Coated Platters stacked on a Spindle.

— Spindle is Responsible for Rotating the Platters.



Physical Components: Platters
 Spindle
 Head actuator arm
 Stepper motor.

Logical Components: Sector
 track
 Cylinder
 cylindrical group

Sector: A Smallest addressable unit on a hard disk is called Sector.

- one Sector can hold 512 bytes of data.
- We cannot increase Sector size. it is Vendor Specific.

Track: Collection of Sectors is called a track.

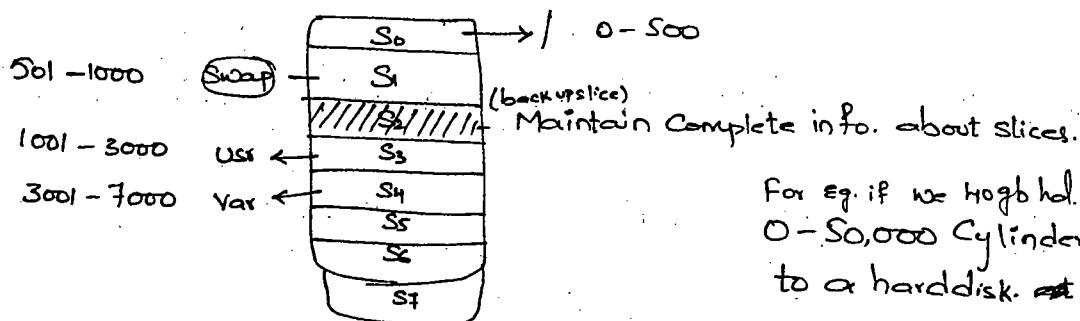
Cylinder: The collection of tracks on one (or) More Platter is called Cylinder.

Cylinder group: Collection of Cylinders.

- one cylindrical group can hold 16 Cylinders.

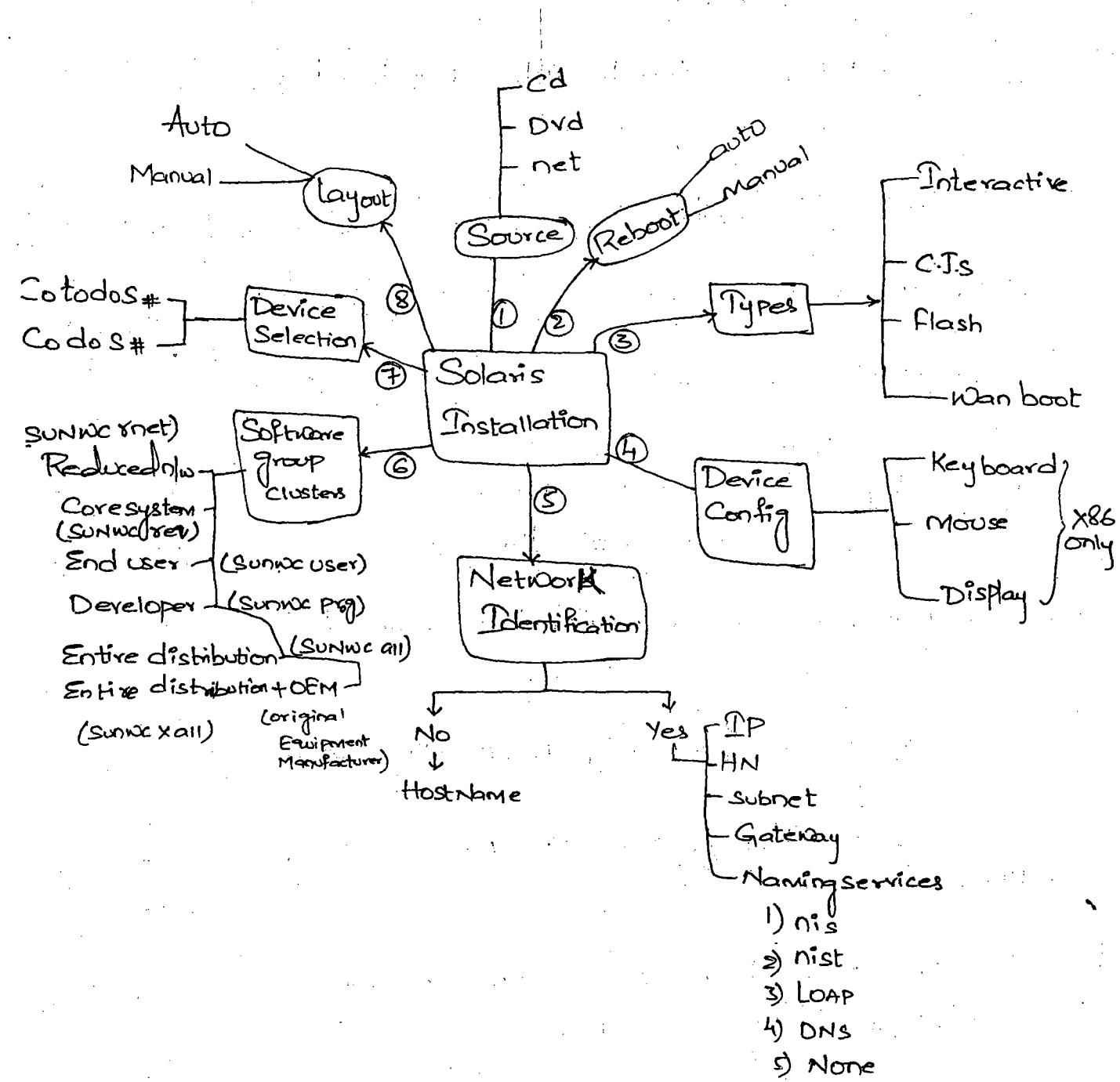
Disk in Solaris:

When We Initialize a disk under Solaris Operating System, disk is going to divide into 0-7 Slices.



- When we try to give same cylinder to more than one slice it will allow but it corrupts the previously allocated slice.
- After installing OS we can create extended partitions. (ss, ss, st)
- At max we can install only 4 operating systems. in a harddisk
- Because on a physical harddisk we can create only 4 primary partitions.
- If we want more than 4 OS then we can depend on 3rd party tools (Vmware).

Solaris Installation Chart



- Package is Collection of files & dir.

- Collection of Packages Cluster Pkg

Sw groups:

Reduced nw → no network connections only execute cmd's.

Core System:
Participate in nw & only act as client not as server
telnet, ftp

End User : (Reduced n/w + Core System)

We get GUI & C.d.e.

Developer:

Entire distribution: we will get all the above. (^{it is a} server)

Entire distribution + OEM : only in Environment level.

we can be in n/w at boot level also.

It is Licensed.

Types of Installation:

Initial (overwrite the o/s)

- Interactive Upgrade (to go from ^{normal ver. to} upgradation version).

- CJS (Custom Jump Start)

- Flash

- Wanboot. (with the help of http Protocol we can install o/s in internet)

- Flash is faster than CJS.

Using flash we can dump entire thing from one system to another.

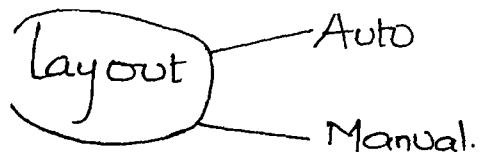
(Ultra60)

(Ultra60)

Here it should have same machine other side.

But

By using CJS we can install in multiple systems at a time.



- If we Select Auto layout it will Create only 3 Partitions.

Root filesystem

Swap filesystem.

/export/home.

- If we Select Manual layout. we have toassing Partitions.

Critical filesystems.

When we Create Manually it is Mandatory to Create Critical filesystems.

Usr

Var

Root

Swap.

If root filesystem is full then we can't access. have to trouble Shoot.

Minimum Requirements for Installing Os :-

12Gb — Harddisk

128mb — Ram (text based)

>128 < 256mb → Console based Installation.

>256 → GUI based.

lbin } Static |usr|bin (exact path)
|sbin

|usr|sbin. (exact path)

etc — Configuration files.

|usr (thix system resources) (all source files/or commands).

|var — dynamic file system

|opt — ThirdParty s/w.

|tmp — Temporary.

We have 5cd's for installation.

1|5cd — Core Operating System.

2|5cd — GUI Related Packages & N/w level packages. (eg: bash)

3|5cd — Acrobat Reader, Dictionaries etc., PDF launchers.

4|5cd — All Additional Packages (eg: Manual Packages)

5|5cd — for additional Languages.

How to Install OS on Sparc Machines? ($\sim 1\text{hr}$, 1/5, 2/5, 4/5)

- Insert 1st Cd

[Stop] + a ↴ then it will goto ok Prompt. ^(bios level)

OK > boot cdrom ↴ (this cmd used for G.U.I based Installation)

OK > boot cdrom -text ↴ (textbased installation)

OK > boot cdrom -nowin ↴ (Console based installation)

- If we go with text based we can perform other operations ^{Console.} i.e., mouse

- Console based we don't get any.

Note:

OK > go ↴ (It will goto os level)

To Come back Press [Stop]+a

How to Perform Installation on X86 Machine?

- Insert 1st Cd & Restart the System.

Demo Installation.

Select type of Installation:

in 30sec if we don't select any option
(by default it takes Interactive)

Select Language : English

1. Solaris interactive

2. Custom Jumpstart.

3. Solaris Interactive text (text)

4. Solaris " " (Console)

Welcome Wizard

[Next] ↴

Networked:

Networked

Non-Networked

[Next] ↴

DHCP for rt1so (network)

Use DHCP for rt1so

- Yes
- No

Next ↵

Hostname for rt1so

Hostname for rt1so

«Anyname»

Next ↵

IP Address for rt1so

↵

Netmask for rt1so

↵

Enable IPv6 for rt1so:

- Yes
- No

Next ↵

Set the default Route for rt1so:

- Detect one
- Specify one
- None

Next ↵

Kerberos

Enable Kerberos

- Yes
- No

Next ↵

Name Service

NameService:

- NIS+ None
- NIS
- DNS
- LDAP

Next ↵

Time zone.

+
2
3

Next ↵

Continent & Country.

→ Asia

└ India

Next ↵

Date & Time

Next ↵

Root Password

Retype

Next ↵

Confirm ↵

Welcome

Next ↵

Installer options

Reboot

✓

Eject

✓

Next ↵

Advanced Features in Solaris 10 OS.

- Zones
- SMF (Service Management Facility).
- dTrace (dynamic trace) (Self healing)
- NFS 4 version
- ZFS (zeta file system)

Specify Media

Media

CD/DVD

Next ↵

Licence

✓ Accept

Next ↵

Select Upgrade/Initial

✓ Initial

Next ↵

Select type of Install

Default ✓ Custom

↵

Select S/W localizations

Just Press ↵

Select Products

Just Press ↵

Additional Products

None

Select Solaris SW group

Entire group

Disk Selection:

Select Coda (boot disk)

Next ↵

Coda

Next ↵

Customize disk Partitions

↵

Preserve Data (If any Previous Yes)

Yes

No

↵

Modify ↵

Disk Coda

<u>Slice</u>	<u>Filesystem</u>	<u>Size</u>
0	/	4000
1	Swap	514
2	/usr	6000
4	/var	1000
5		
6		
7		

(by default it should have 3.5gb)

Apply **OK**

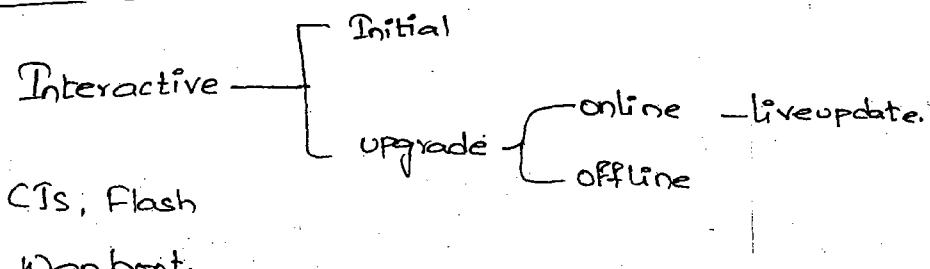
Confirm

Next ←

Install Now ←

9/10/09

Types of Installation:-



#luupdate

-to install online

loadvtoc:-

Previous Partitioned info. of hard disk.

Locations for Installation logs :-

| Var | Sadm | install | Logs

- used to maintain all log information.

- If we successfully install 2 cd's ^{in x86}, it generates 4 log messages.
- In Sparc it generates 2 log messages.
- Apart from 4 log messages if we see any other messages then it is not successfully installed.

Cat | Var | Sadm | System | admin | CLUSTER ↴

O/P: SUNwc all.

- displays what cluster is installed (S/w group)

If we want to know how many S/w groups installed.

Cd | Var | Sadm | System | admin ↴

Cat . / clustertoc | grep -i meta ↴

lists all Cluster Packages.

- To know how many Partitions are installed, and Complete installation logs.

Cd | Var | Sadm | install - data ↴

Cat install - log ↴

To Know what are the Packages Installed.

```
# Cd /var/sadm/pkg
```

```
# ls
```

File System :-

- In order to access the hard disk we need to Create a file system.

- It's a Collection of files & directories.

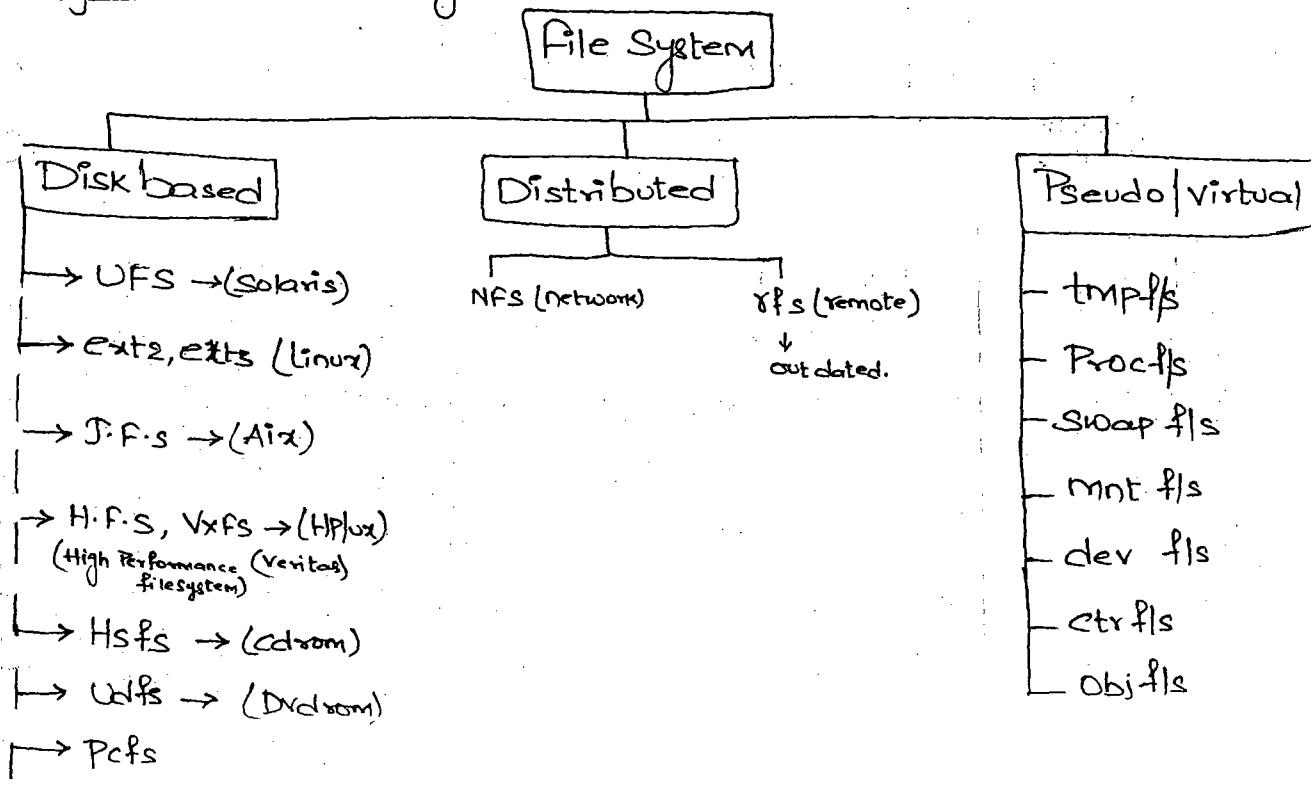
defn: It's a Collection of Control Structures which provides a disk to Store, Retrieve, Manipulate the data on a hard disk in Pre-defined Manner.

The default file system in Solaris is UFS (Unix file system).

UFS is a block based file system.

format
Will allocate
file sys. hierachy

Types of File Systems:



UFS :- (Unix File System).

- It's a default file system in Solaris.

Location for UFS filesystem:

```
# vi [etc|default|fs] ↓
```

- Responsible for maintain default file system.

OP:- LOCAL = UFS.

j: Ss — 5GB

[dev|ldisk|c0t0d0s5]
Logical Path.

Now If we need to access the Ss ~~file~~.
We need to provide Mount Point.

↓
dynamic dir.

Mount Point is used to access the f/s.

```
# Mount -f ufs [dev|ldisk|c0t0d0s5] ↓  
- /m.p
```

- We cannot give multiple fs..

- If we want give any filesystem (-f ufs) then it will search
in default ^{file} system. If

hsfs :- (high Sierra File System)

- Responsible for access cdrom type of devices.

```
# Mount -f hsfs <cdrom devicepath> /M.P ↓
```

Cdfs is filesystem
in windows

iostat is used to get harddisk & cdrom paths, floppy, Remote Paths.

iostat -en ↗

Udfs: (Universal disk format file system)

- Used to access dvd rom devices.

Pcfs: (Personal Computer file system).

- In order to access any DOS formatted device (or) Removable storages we need to create Pcfs.

Eg: Floppy's, Pendrive.

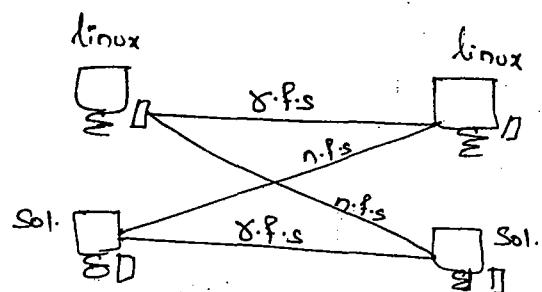
Volcheck If we need to get Pendrive Path.

Distributed File System:-

- N.F.S is used to share files & dir. with diff. users.
- X.F.S won't support to share files with different O.S. but N.F.S supports i.e., N.F.S supports heterogenous platform.

N.F.S

X.F.S (outdated from Sol. 2.x)



uname -a — displays version of OS.

cat /etc/release ↴

- to know the release version of OS

Default NFS Version now is 4.0

Pseudo Fis:

Generally These type of filesystem are used to increase the System Performance.

/tmpfs:- is responsible for maintain /tmp directory.

- /tmp directory stores temporary info.
- once we reboot it will be empty.

/Procfs :- Used to maintain /Proc directory.

- /Proc directory used maintain all active Processes info. in terms of directories.

```
# cd /Proc
```

```
# ls -l
```

```
128 1982  
148
```

- All the active Process id's are stored in /Proc directory as directories.

Ps -ef to get all active & sleeping Processes info.

/Swapfs:- is a file responsible for maintain Swap Memory (Virtual Mem.) on a disk.

/Mntfs:- Responsible for maintain "/etc/mnttab" file.

- It's a Read only file. Even administrator also can't edit this file.
- When we execute Mount (or) Umount Commands /etc/mnttab will get updated.

— It is also responsible for maintain mounted partition information.

|devfs :- Responsible for maintain /dev & /devices directory

— maintain currently installed device drivers.

|objs :- (Object files)

— Responsible for maintaining currently installed modules (device drivers).

|System|Object

device
— updates the drivers. We find info.

|ctrfs :- (Contract) — Responsible for maintaining the relation b/w Process & Processor.
(Child & Parent Processes).

Ptree

Used to list multiple threads info.

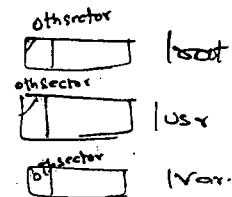
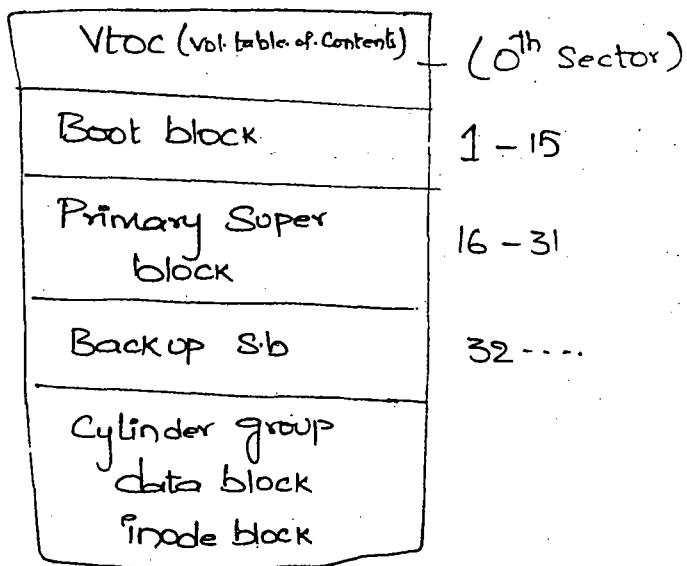
Prstat

↳ List Zombie Process Status.

Zombie Process ~~status~~ ^{Convention name} is <defunct>

Logical Components OF UFS :-

- Control Structures are nothing but logical components.



Vtoc :- (Volume table of Contents)

- Responsible for Mounting entire Slice / Partitioning info. in terms of Sectors.

Prvtoc — used to display Volume table of Content.

Syntax:

Prvtoc <raw path of slice>

e.g. # Prvtoc /dev/rdsk/C0T0D0S# <

It resides 0th Sector.

Boot block :- It resides 1-15 Sectors

- Always boot block is activated in Root Slice only.

Bootstrap is ^{Program} used to load files from Kernel → Ram.

- It contains Bootstrap Script, which contains all the Module info, Kernel swaps this info to ram.

Boot block Location:-

```
# Cd /usr/Platform/`uname -m`/lib/fs/ufs <br>
# ls
bootblk.
```

Restoring the boot block:-

```
# installboot bootblk <Specify Path of slice>
```

Re-install the bootblk. in /root slice.

Eg: installboot bootblk /dev/sdsk/C0t0d0s0

In x86:

```
# installboot pboot bootblk /dev/sdsk/C0t0d0s0 <br>
# reboot.
```

Primary Super block:- It Resides 16 - 31 Sectors.

- It maintain Critical info. about the
filesystem (slice).

file system size

Used & free datablocks & i-node

Mount point name

Cylinders & C.G

table info.

} Critical info.

format in windows.

newfs is used to allocate datablocks to file system (slice).

With the help of Backup superblock we can restore Primary Super block.

Restoring Primary Superblock :-

to restore we have to know how many backup superblocks are there.

newfs -N /dev/rdsk/c0t0d0s5

This command is used to list filesystem parameters.

fsck -F Ufs -o b=32

— to restore Primary Superblock.

option
backup
super
block
copy

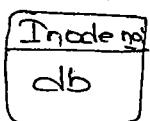
Max. it can have only 10 backup superblocks.

Cylindrical Group :- Collections of Cylinders. One Cylindrical group can hold 16 cylinders.

Data block : It is a storage area. The default datablock size for files & directories is 2kb.

inode block :

filename.



Inode no.: Except file name each and every info. of file is maintained in inode no.

Parent directory maintains filename.

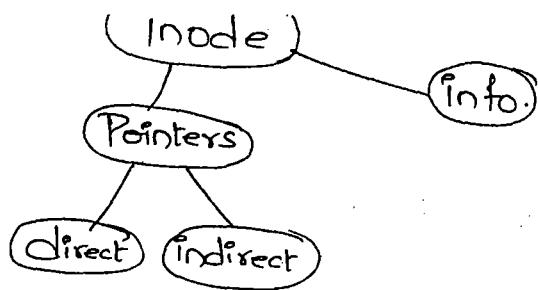
Inode Maintains: filesize

links

owner, group

Access time

Permissions



file System Parameters :-

Physical Block

Logical Block

Fragment

Min free Space.

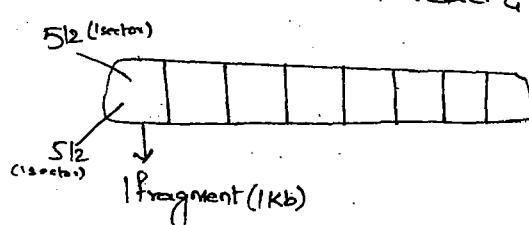
Physical Block :-

Size is 512 bytes. (1 sector)

Logical Block :-

Size is 8kb -(8192 bytes)

Controller can read & write using 8k



Note:- In UFS we have default defragmentation (compress) facility

1 Fragment (1 kb)

Smallest portion in logical block is fragment.

Min free space: When we are creating files. Some amount of space is reserved for future purpose that is called Minfree Space.

- default minfree space is 10% for any file.

```
#fstyp -v /dev/rdisk1 | grep -i minfree
```

- used to list minfreespace.

```
#tunefs
```

- used to utilize minfree space.

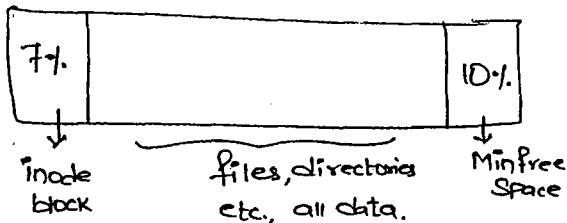
It is used to increase & decrease minfreespace.

```
#tunefs -m 5 /dev/rdisk1 | cat > /etc/fstab
```

to specify size

10109

In a slice by default



fsck : (filesystem checking) :-

- fsck is a command used to check file system consistency.
- When we run fsck on a single filesystem, it tries to check filesystem consistency by reading sector to sector.

In which cases we need to run fsck?

- due to improper shutdown.
- If cylinders are overlapped.

Generally we need to run fsck command in a single user mode.

We have 5 phases in fsck.

- 1) Checks blocks & sizes.
- 2) Checks connectivity

3) Checks Pathnames

4) Checks Reference Count

5) Checks Cylinders & Cylindrical groups.

Q 1 - Allways fsck checks from absolute path only.

Logging: for example we have a file of size 5GB & let us suppose 2GB is filled & 3GB is empty. When we run fsck command it will check each and every sector of entire 5GB. In order to check only the filled content we use logging concept.

Q 2 - When we create any filesystem by default 'lost+found' directory is created. This will be helpful when the system is shutdown improperly to retain the lost data.

Q 3 - Always we need to run fsck cmd. on Rawpath of the slice.

Syntax:

```
# fsck <options> <Rawpath of the Slice>
```

Eg: # fsck -y /dev/sda1

↓
PreAnswering

To Check whether the filesystem is Consistent / Inconsistent:

```
# fsck -m /dev/sda1
```

O/P: OK

If we get any other message other than OK then it is inconsistent we need to execute fsck.

Files Related to File System :-

- /etc/vfstab
- /etc/mnttab
- /etc/format.dat

* /etc/vfstab :-

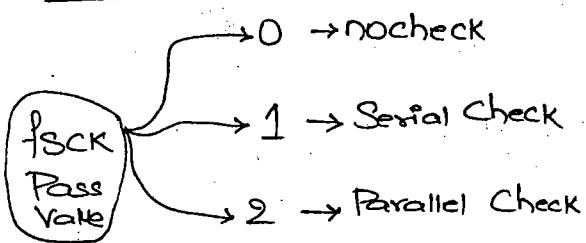
Responsible for Maintain Permanent Mount Points

(or) Permanent file system attributes.

vi /etc/vfstab ← we have 7 fields.

①	②	③	④	⑤	⑥	⑦
Device to Mount	Device to run fsck	Mount Point	File System type	fsck Pass value	Mount at boot	Mount Options
dev/dsk/c0t0d0s5	dev/dsk/c0t0d0s5	/sun	ufs	1	yes	-

fsck Pass Value:



If we give any improper entries under /etc/vfstab then automatically placed in Maintenance Mode.

We need to provide 1 tab space for each & every field.
(Mandatory).

* /etc/mnttab:

Responsible for Maintain all Mounted Partitions information.

- It's a Read only file even admin can also can't edit this file.

ls -l /etc/mnttab ↴ (It's only read only file).

O/P: -r--r--r--

* /etc/format.dat :-

Responsible for Maintaining disk Partition Information

Eg: # format ↴

- format Cmd will internally reach /etc/format.dat file.
- Even we can customize this file.

Commands Related to File System:-

* format

* Newfs

* Mount

* Mountall

* Umount

* Umountall

* Prvtoc.

format :- is the only Cmd. used to display Hard disk Physical name & logical names.

format ↴

Cat to do Pen @1,2,3&...

↓
Logical
name

↓
Physical
name.

- With the help of format Command we can

* Create/Remove file

* Label disk.

- * Low level format
- * Check d.blocks Consistency
- * Customize /etc/format.dat

Newfs :- Newfs is a command used to build a file system.

Syntax:

```
# newfs <options> <rawpath of the slice>
```

Always we need run newfs on Char. Spi. file only.

<options>

N → filesystem Parameters

m → Min free space

i → inode size.

j: # newfs -N /dev/rdsk/C0t0d0s5 ↵

newfs -m /dev/rdsk/C0t0d0s5 ↵

to give 30% min free space

newfs -m 30 /dev/rdsk/C0t0d0s5 ↵

The default inode size is 2kb.

If we want to reduce the size of inode.

newfs -i 1024 /dev/rdsk/C0t0d0s5 ↵

Mount:-

Mount is a Process of attaching a directory to access point to file system.

Syntax:

Mount <options> -o <options> <Block Special Path>

<Mount point Name>
(or)
dynamic dir, Name

Options 1>

-f → used to specify file system.

Options 2>

- ro (read only)
- rw (read write)
- Logging (synchronous) → default.
- noLogging

Mount -f ufs -o ro /dev/dsk/c0t0d0s5 /Sun ↴

Mountall :- Used to Mount all Partitions stored under /etc/vfstab file.

Unmount:- is a Process of detaching an access point to file System.

"df": to know disk size / fs size
display

"du": Space used

Learn:- After Installation.

Go to Terminal

`cat /etc/release` :- Responsible for maintaining OS release version

`cd /var/sadm/install/logs` ↴ : to know how many cd's we installed.
ls

`cd /var/sadm/system/admin` ↴ : displays entire Cluster details
cat CLUSTER

`# cat /etc/cluster | grep -i meta` : display Meta info.

`# cd /var/sadm/Pkgl` ↴ : displays all Package names
ls

`# cd /var/sadm/install_data`
ls
More install_log : Maintains all installation logs.

`cd /usr/bin
ls *sh` : displays all shells.

`echo $SHELL` : to display Present working shell

cd /dev/dsk
cd /dev/rdsk

: Physical names (block spl. files)
: Logical names (Char. spl. files).
Physical

iostat -en

: displays the S/W & H/W Errors.

cfgadm -al

(Work only on Sparc Machines)

errors path

/etc/init.d

Sendmail

Demo for File System :-

df -h

: Used to list mounted Partitions information in Mb & GB

Filesystem	Size	Used	Avail	Capacity	Mounted on.
------------	------	------	-------	----------	-------------

df -K

: in Kilobytes.

df -n

: used to display filesystem type.

fstyp /dev/dsk/c0d0s4

: used to display filesystem type of Particular Slice.

To Create a file we need to check how many harddisks are there.

devfsadm -c disks

: it will not display D.P.

format

Specify the disk no: 0

Format Menu.

format > p ↵ : display the Partition table.

Partition > p ↵ : displays the slices info.

Partition > 5 ↵ : we can build the PIs in 5th slice.

Enter Partition id tag [unassigned]: ?] display which file
we can create.

Enter Partition id tag [unassigned]: alternates

Enter Partition Permission flags ^[w]: ?

Enter new Starting Cylinder : 264 ↵

Enter Partition Size : 1g

Partition > p ↵ : displays Partition table.

Partition > l ↵ : for making Partitioning table Permanent.

Ready to Label disk, Continue? Y.

Partition > q ↵ quit from Partition

format > q ↵ quit from format.

Creating The File System :-

mount -dsk
newfs -r dsk

newfs /dev/rdsks/Codoss ↴

↳ newfs: Construct a new file system /dev/rdsks/Codoss : ↴

Now we have to Create Mountpoint.

After installing file system

mkdir /Sun

mount /dev/dsk/Codoss /Sun ↴

to check mounted (or) not

df -h

After mounting we need to save filesystem attributes permanent.

vi /etc/vfstab ↴

goto last line.

/dev/dsk/Codoss /dev/rdsks/Codoss /Sun ufs 1 yes -

How to increase a Particular f/s size if it gets filled up. (online)

format

P ↴

P ↴

G ↴

↴

starting cylinder [700]

Size [4GB] : 10GB

P ↴

D ↴

df -h ↴

0 - 500
500 - 1000
1000 - 7000
7001 - 8000

growfs ↴: used to increase f/s size online.

growfs -M users /dev/rdsks/Codoss ↴

rawpath

Newfs for new f/s. to grow the f/s size growfs.

64

How to delete a Partition?

umount /sun ↴

vi /etc/vfstab ↴

go to lastline & delete the path.

format ↴

↳ format > P

↳ Partition > P

↳ Partition > S

Enter new starting cyl [1] : 0

Partition > P

Partition > l

Partition > q ↴

format > q ↴

How to See The Min free Space?

fstyp -v /dev/dsk/c0d0s5 | grep -i minfree.

61.

to make 2-1.

tunefs -m 2 /dev/dsk/c0d0s5 ↴

News - i

to reduce the inode no's size.

`#df -e` — displays empty inode no's

Prvtoc.

Dual Booting:

Lab on Project:1

My Computer → Right click → Manage

Disk Management (double click)

Delete drives (d; e:)

12/10/09

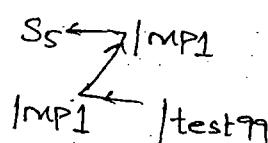
Lofs: (loopback file system)

- Used to provide multiple access points to an existing file system.

`#mkdir /test99 ↴`

`#mount -f "lofs" ↴ /mp1 ↴ /test99 ↴`

`#df -h`



* directly we cannot unmount the /mp1 directory first we need to unmount /test99.

(S5)

Imp1

```

  └── F1
      ├── F2
      └── F3
  
```

(S5)

Imp2

```

  └── index.txt
      └── task.temp
  
```

Unmount Imp1

cd Imp2 <

ls <

f1 f2 f3

Mkdir Imp2

Unmount Imp2

ls

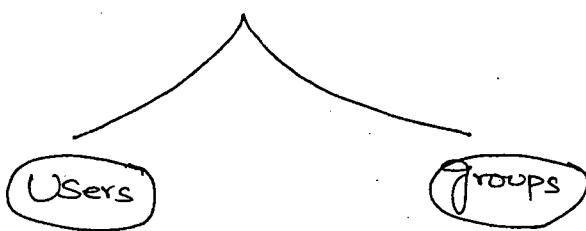
cd Imp2

ls

index.txt

task.temp

ACCOUNT MANAGEMENT :-



- In Order to access System (or) Server Resources We need to Create User ac.

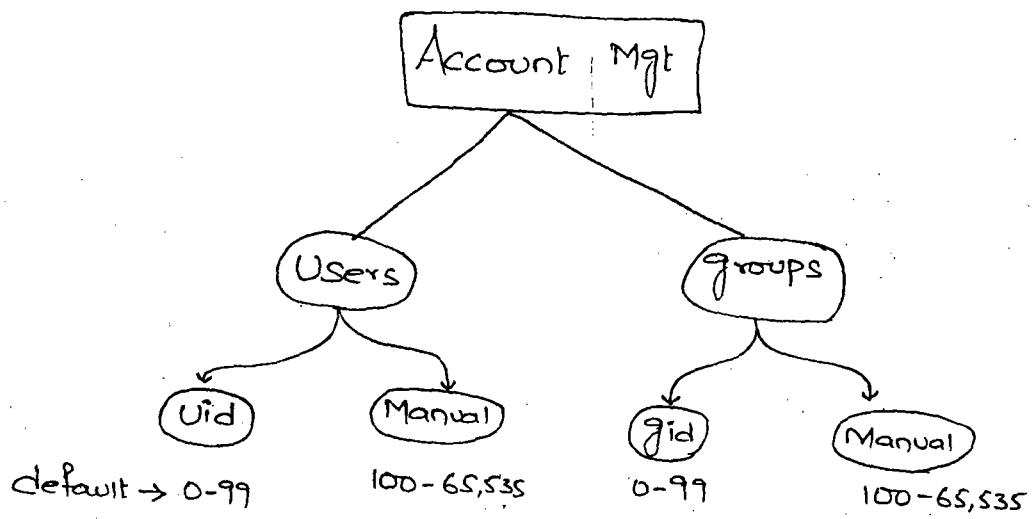
- Root is a by default User ac.

- We Can Create User's and group ac, in 3 Ways.

1) Cli ↗ Useradd
 ↗ groupadd

2) Gui ↗ Admin tool (Sol. 9)
 ↗ SMC (#SMC 8) - (Sol. 10) (4/4 CDs need to be installed)

3) Shell Script: - 2 or more users at a time.



- Based upon the Project (or) requirements we need to Create Multiple User Alc.

- Root is default for root Alc

- by default 15 groups ^{have} are been created.

- One User can participate at max 17 groups.

17
 16 secondary 1 primary

Commands for Creating Groups:

groupadd

groupmod

groupdel

dispgid

File Related to group:

/etc/group

How to Create a Group?

Syntax:-

```
# groupadd <options> <G.name>  
L -g → gid
```

System can identify groups with gid's

Admin with gnames.

Eg:-

```
# groupadd -g 137 unix ↴
```

```
# dispgid
```

 — displays groupid's.

```
+ Vi /etc/group
```

Gname : G.Password : gid : Secondary group : Username.

```
# groupadd <groupname> ↴
```

 — System will provide groupids.

```
# groupmod
```

 — to modify group attributes.

```
# groupmod -n <newgroupname> <oldgroupname>
```

```
# groupmod -g <newgroupId> <oldgroupId>
```

Deleting a Group :-

```
# groupdel <gname> ↴
```

To see no: of users in a group

```
# logins -g <g.names>
```

User Attributes:

Uid ↗
100 - 65,535
'0' - default root uid.

Uname (unique name)

Shell (bourne shell by default)

Homedir

Comment

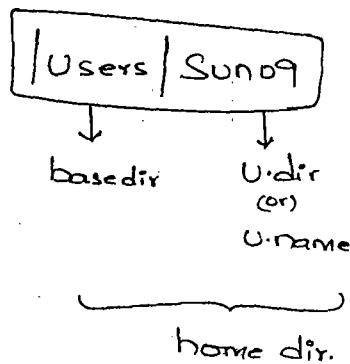
Shell

— Root Userid always 0.

— even we can duplicate Userid also.

— default shell for all users is bourne shell. Even we can customize default shell.

Home dir:- After logging in as a normal user the files & dir. which we create will store under home dir.



Comment: for identification we give Comment.

(to identify what type of user).

Skell: When we are Creating a User a/c. all the files from Skell directory are inherited to Normal User a/c.

/etc/skel - Contains the default files like
- bash_history
- bash_profile

We call these files are Login Initialization files

banner - to get banner msgs.

/etc/skel is responsible for maintaining login initialization files.

Files related to User a/c:-

/etc/Password

/etc/Shadow

/etc/default/Password.

/etc/Password:

Note: When we create a User a/c it automatically updates in /etc/Password & /etc/Shadow files.

- This file is Responsible for Maintaining User info.

/etc/opasswd - is a backup file for /etc/Password.

Attributes: (7 fields)

① Uname : ② Passwd : ③ uid : ④ gid : ⑤ Comment : ⑥ Homedir : ⑦ Shell
link

/etc/shadow:- This file is responsible for maintaining all users password info. in encrypted format.

* It should store in only 13 char. in Passwd.

This uses Unix algorithm.

attributes: (9 fields)

① Uname : ② Encrypted : ③ Lastchange : ④ Min.no. of days to change Password : ⑤ Max no. of days to change Password : ⑥ Warn : ⑦ Inactive : ⑧ Expiry : ⑨ Future purpose.

When we change the Password only first 3 fields are getting updated.

Commands Related to User:

Useradd

User Mod

Userdel

dispid | list users

Useradd :-

#Useradd <options> <Username>

U → uid
G → Primary group
G → Secondary group

-c → Comment
-s → Shell
-K → S~~kell~~
-d → Homedir
-e → expiry
-m → assign home dir.

useradd -u 148 -g unix -G dba -c "normaluser" -s /bin/bash

-K dir. -d /usr/sun -m sun ↓
↓
username

Passwd sun ↓

New:

Retype:

listusers ↓ to check User ale is available (or) not.

How to login as a normal User?

telnet ? ↓ used to login as a local user.
* su -

Login: sun ↓

Password: xxx ↓

\$

useradd -d /users/sun -m Sun ↳ - it will take other options by default

useradd Sun ↳

/export/home - default home dir. for normal users.

- default group for all the normal users is Other.

- default group for root users is Root.

- default group for root users is Root.

Usermod:

Syntax:

usermod <options> <username> ↳

-l → change username
* represents
→ all options.

usermod -l <newusername> <oldusername>

Userdel:-

userdel <username> ↳

- only deletes the user acc. but the files & dir. are available in home dir.

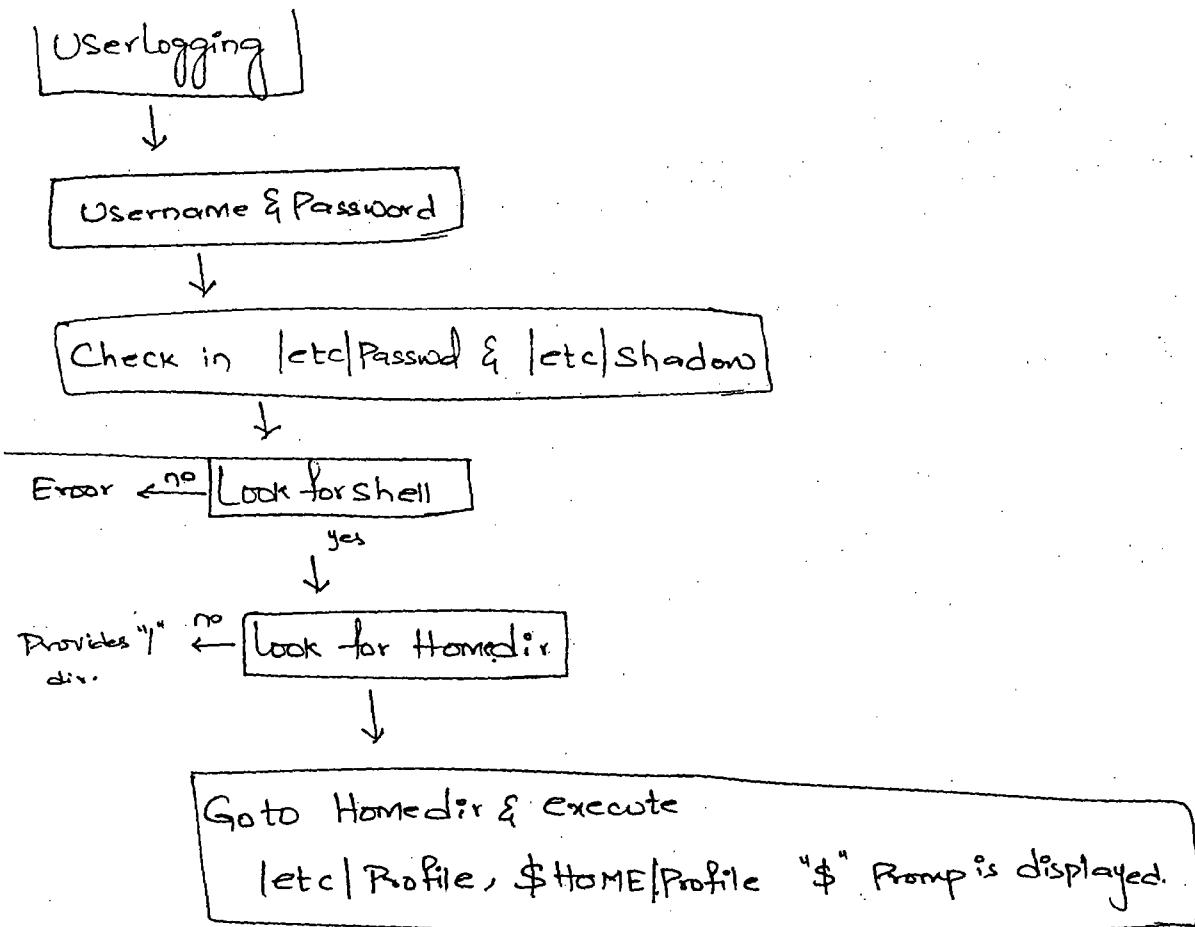
userdel -r <username> ↳

- deletes recursively entire files & dir. created by that user.

/etc/default/passwd:

Responsible for maintaining User Password.

- Length
- Min
- Max.
- alpha char
- number



Password States:

3 States

NP → No Password

LK → Locked

Ps → Password existing.

List all users who don't have Passwords.

login -P

Syntax:

# Password	<options>	<username>	↓
	-S	→ to see the status	
	-L	→ lock	
	-n	→ Min. no. of days to change Password.	
	-X	→ Max " "	
	-U	→ unlock	
	-W	→ Warn days	
	-P	→ Inactive	
	-d	→ delete Password.	
	-e	→ Expiry date (mm\dd\yy)	

Note: By default User alk's Password is locked status.

for unlocking

Password -U Sun ↓

↓
Username

for locking

Password -L <username> ↓

for deleting

Password -d <username> ↓

Password -n 2 -x 4 -w 3 Sun ↓

Once the Max. no. of days to change Password is completed then alk will go into inactive status.

The Expiry date format is

mm/dd/yy

usermod -e

Demo:

Creating Groups:

at /etc/group by default 19 groups

at /etc/Passwd by default 15 users.

groupadd -g 648 unix

groupadd -g 649 dba

groupadd -g 650 dotti

dispgid — display available groups.

Creating Users:

useradd -u 137 -g unix -G dba -c "Normaluser" -s
/bin/bash -d '/usr/User1'
-m User1 ↴

cat /etc/Passwd | grep User1

Passwd User1

listUsers

`# [id -a User]` ↳ displays Primary group & Secondary group.

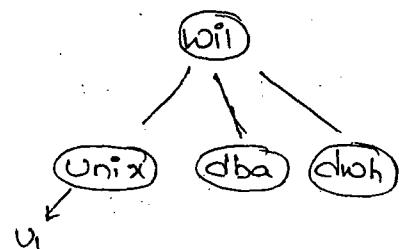
~~#~~

— Check out /etc/shadow file Permissions.

date ~~010120091230~~ to change date
010112302009
↓ ↓ ↓ ↓
day hrs min year

13/10/2009

`# usermod -G dba,dish vi`



- Based on Setuid, Setgid Users can give Password.

- As /etc/shadow file have Read only Permission to users also. But Still we can write using Setuid & Setgid Variables.

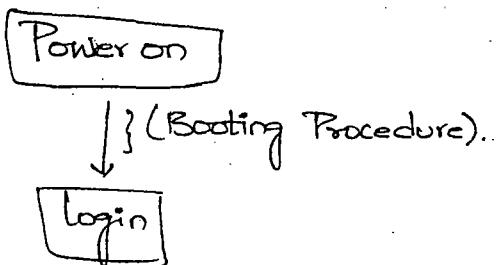
`# PwConv` ↳ is used to synchronize the info. to /etc>Password to /etc/shadow.

- After executing this cmd. once again assign the Password.

How to Duplicate the Usrid ?

```
# usermod -U 0 -d /usr/u1 -M u1
```

Booting & Shutdown :-



The first Phase in Booting Procedure is

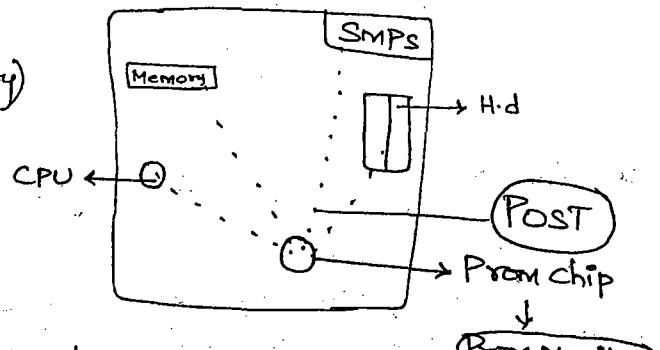
POST : (Power On Self Test)

(Sends signals to each & Every hw comp)

PROM : (Programmable Read Only Memory)

Prom is Responsible for running the Post.

- Each & Every Sparc Machines are having 1mb Prom chips which is typically located on Main board.



In Older Versions we have Pluggable Prom chips.

- The Soldered Prom chips are available in New Versions.

↓
"F-PROM"

Windows → bios
Solaris → PROM

Stop + a

OK >

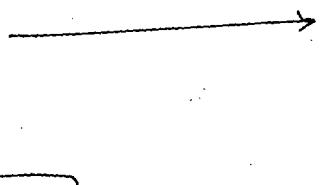
↓
O.B.P (Open boot Prompt)

- In Solaris 10 we have different type of booting Procedure
Phases.

Startup & Shutdown

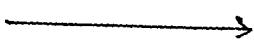
Boot Process

Boot PROM Phase



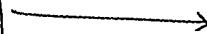
PROM runs POST
Boot Locates boot device
Boot reads & loads Primary boot blk.

Boot Program Phase



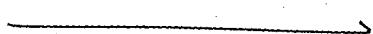
Primary bootblk loads Secondary boot block i.e., Ufsboot
Ufsboot loads Kernel (32-bit
64 bit, Platform independent,
Platform dependent)

Kernel Initialization Phase



Kernel initialize itself &
loads Modules, reads
etc/system

Init Phase



Kernel reads /etc/inittab
Start the Master Daemon

Start Phase
(SVC-startd)

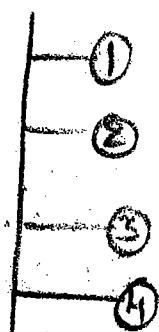
Responsible for Start & Stop all the Services.

Power on → POST →

↓
(On Screen) Banner msgs

↓ —> Boot Promt Line
Memory Size (ram)
↓ Machine Arch.
↓ Keyboard
↓ hostid

Hhd



OK > auto-boot ? - to check which harddisk has bios
OK > (false) (true)

OK > boot-device - based on the settings it will boot.
disk① disk②

cdrom net

find bootblk

lusr / Plat / uname -m / ...

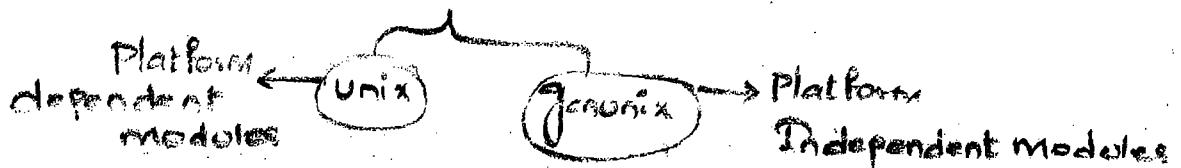
here it finds boot block

If it doesn't find we can get it by going to **install boot**.

B. Program phase → Secondary Boot Program



(UF's boot)



When secondary boot Program is Started Each modules are loaded from disk to ram. This Procedure is Called a bootstrap Program.



After Secondary Boot Program → Looks for Kernel.
(64 bit (or) 32 bit)

Kernel is of two types

- Monolithic (Old one, it tries to load needed & not needed modules also)
- Modular (It loads only needed modules).
(to reduce the boot time)

③

Kernel Initialization : Kernel initialize itself by reading Phase

[etc/]/System file.

→ responsible for maintain Kernel tunable Parameters.

→ Try to start init Process,

INIT Phase

The first Process in system is Init Process.

- /etc/init.d → Appropriate runlevel

③ svc.startd: (only sol.10)

all the services, h/w level Services will start at this phase.

↓
Console login.

- In Sol. 10 we have 5 booting Procedure Phases.
- In Sol. 9 we have only 4.

Run level's:- (0 - 6)

init: is a cmd. used to change one runlevel to another runlevel.

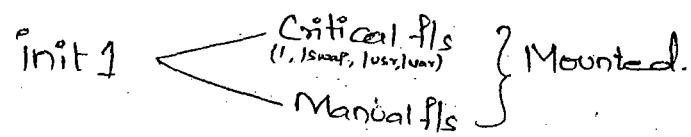
Runlevel 0: (#init 0)

- It brings the System into O.b.p level (bios)
- Being in runlevel 0 we can interact with h/w devices with the help of Rom Variables.

Runlevel 1: (#init 1):-

- The System will be Placed in Maintenance Mode. Being in Maintenance Mode we can access Critical PIs as well as Manually created fIs also.

- Generally we need to install Patches being in this runlevel.



- All the backup's are taken in Maintenance Mode only. (Single UserMode).

Runlevel S or (init 5) :-

It is a Single Usermode. Only Critical f/s are in Mounted Status.

Runlevel 2 : (#init 2) :

- Multiuser mode without nfs.

- We can act as only client

- All the nwo related Services are UP.

Runlevel 3 : (#init 3)

- is a D.R.L (default runlevel)

- it is multiuser with nfs

- A Complete Server.

Runlevel 4 : (#init 4)

- We can customize this level for future purpose.

Runlevel 5 : (#init 5) Poweroff the System.

↓
OK > Power-off

Runlevel 6 (or) (#init 6):

— restart | reboot

(or)

#reboot

↓

\$

Graceful Shutdown :-

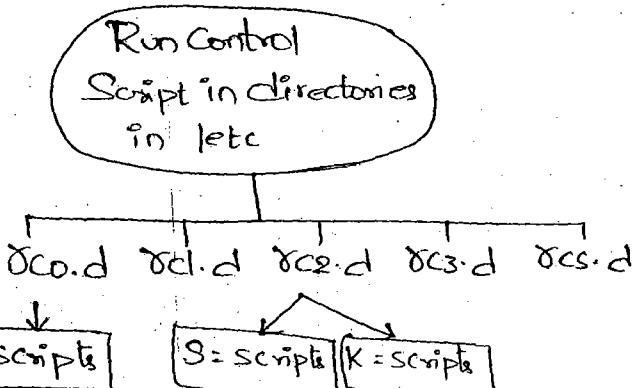
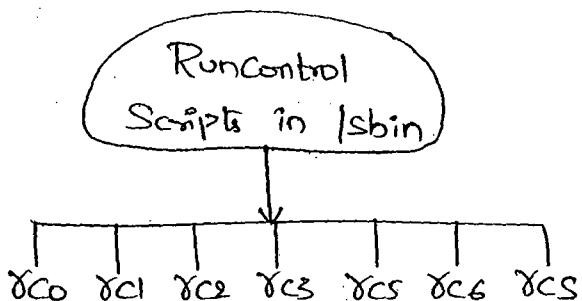
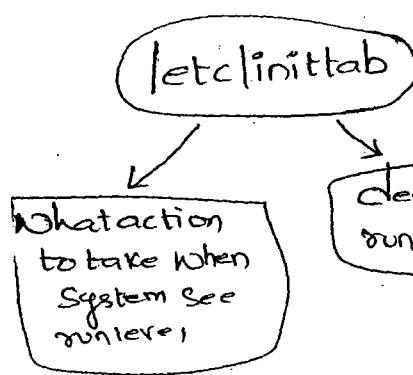
#Shutdown -y -90 -16 ←

↓ ↓ ↓

Seconds Runlevel

Pre-answering

Run Control Scripts :- (RC Scripts).



K — Killing script
S — Starting script.

Where to go ↓ Where you are	0	1	2	3	6
0		boot -s (Login as root)	boot -s #init 2	#boot_d DRL	
1	#init 0 @@ halt		#init2	#init3	#init6 #reboot
2	#init0 #shutdown -g 300 -i0	#init1		#init3	#init6 #Shutdown -g -i
3	#init0 #shutdown -g 300 -i0	-do-	#init2		

SMF (Service Management facility) :-

- It is an advanced feature in Solaris 10.
- It Provides Centralized Configuration information for Managing the System Services and the interaction between one Service to another Service.
- It has Structured Mechanism facility called F.M.R.I (Fault Mgt. Resource Identifier).

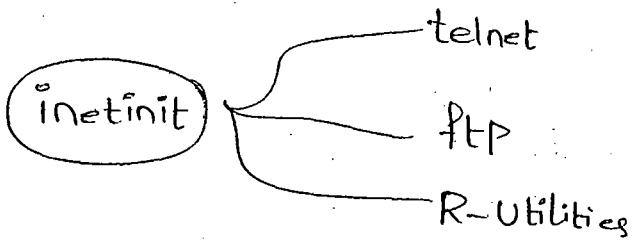
F.M.R.I

```

graph LR
    FMR[F.M.R.I] --- Category[Category]
    FMR --- Service[Service]
    FMR --- Instance[instance]
  
```

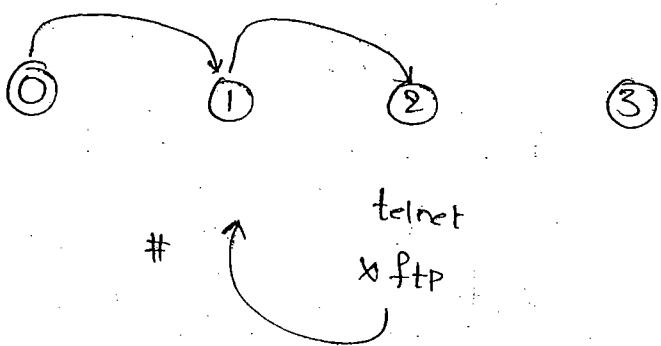
Disadvantages in Solaris 9:-

- We Cannot check Service Status.
- Can Perform only 2 operations for Services i.e., Stop & Start.
- It Won't Provide log Messages for any Services.
- We have only one Main network daemon **inetinit**.



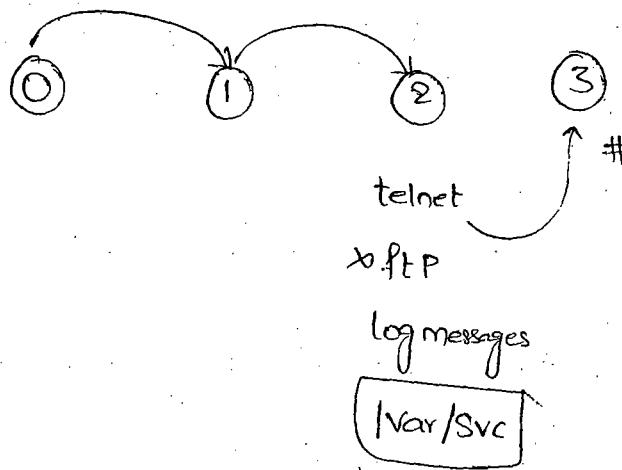
- All these Services are under Control of lrc (legacy run Controls) in Sol. 9.

etc/init.d



If any Service request not running it will be in maintenance mode only.

Solaris 10:



It can automatically heal if ftp doesn't work. It tries to enable. This process is dt-trace.

Service Identifier:

- Each instance of a Service within SMF has a name
- which is referred to as Service Identifier.
- This Service Identifier is in the form of FMRID.

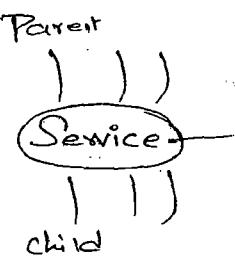
Service Categories: We have different types:

- Application
- Device
- Network
- Milestone
- Legacy
- Platform
- System etc.,

1/10/2001

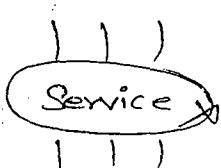
Service States:

1. online: Service is enabled and it has successfully started. that means all dependency & dependent services are up and running.



2. Offline:

Service instance is running but service is not running.



3. Disable: Service instance is not enabled but service is running.

1. LRC: (Legacy Run Control): It is not managed by SMF but we can observe the service.

2. Uninitialized: This state is initial status for all the services before their config. is read.

3. Maintenance: The service instance has encountered an error. That must be resolved by Admin.
Note: due to improper configurations.

7. degraded: The Service instance is enabled but it is running with a limited capacity.

Milestones:

- In Sol. 10 release onwards all the runlevels we call as Milestones.
- A milestone is special type of service which made up of defined set of other services.

Types of Services Milestones:

Single User → Runlevel [0/1]

MultiUser → Runlevel [2]

Multi-User-Server → Runlevel [3]

N/w

- By default milestone is Multi-User-Server. Even we can customize default milestone also.

Service Configuration Repository:-

The Repository db. stores the info.

about the state of each & every service instance.

- It maintains the configuration info. for the services.

- The location for repository db. is `/etc/svc/repository.db`

- We can restore the corrupted depository db also being in maintenance mode.

```
#rm /etc/svc/repository.db  
#reboot
```

goes to Maintenance Mode.

```
#cd /lib/svc/bin  
# ./restore_repository  
#Sync.  
#reboot.
```

- The repository maintained by **Svc. Configd** daemon.

Commands:

- **[#Svcs]** → to see Service Status
- **[#Svcadm]** → to Perform Services administration
- **[#Svccfg]** → to delete a service | import a service.

- If any service is deleted still we can get back the info. by importing the XML Scripts.

[# Svcs -q] : Used to list all Services & Service State & time Stamp.

Eg: enable 11:30 svc:/network/login/telnet:default

#Svcadm enable <FMRI> : Used to enable the disabled(1) Service.

Sol. 9

```
# cd /etc/init.d  
# ./nfs server stop  
# ./nfs server start
```

Sol. 10

```
# svcadm enable nfs/server ↴  
          (or)  
          restart  
# svcs -a | grep nfs/server ↴  
online.
```

Svcs -a will list disable services also.

`# Svcs -d <fmri>`

: used to display dependency services (child)

`# Svcs -D <fmri>`

: used to display dependent services (parent)

`# Svcadm enable -x <fmri>`

: used to enable the service recursively.

If any one Particular Service is not running Properly we can check

`# Svcs -x <fmri>`

We can monitor each & Every Service

`# tail -f /var/adm/messages`

: display last recent 10 message

`# Svcs -l <fmri>`

: used to list Complete info. Regarding the Service.

```
# Svccfg delete nfs|server
```

- To delete a service.

All XML Scripts are located under

```
# cd /var svc/manifest/network/ifs
```

```
# Svccfg
```

```
> import Server.xml
```

```
> quit
```

```
# Svcadm restart nfs|server
```

```
# Svcs -a | grep mile
```

```
# Svcs -a | grep file
```

```
# Svcs -a | grep inetd
```

```
# Svcs -a | grep network|physical
```

} execute these cmds.
if atleast one cmd
doesn't execute then
we won't get GUI Mode.
we need to trouble shoot

How to Set default MileStone?

```
# Svcadm milestone -d Single-User
```

disable → enable

offline → disable
enable

Maintanence → Check the main config. file.
Clear.

Package & Patch Management :-

Package Administration :-

- Pkg. is a Collection of files & dir.

Windows → S/w

Sol. → Pkgs.

Linux → R.PMs

AIX → S/w bundles

HPUX → S/w bundles

- All the Packages are Starting with naming Conventions "SUNW".

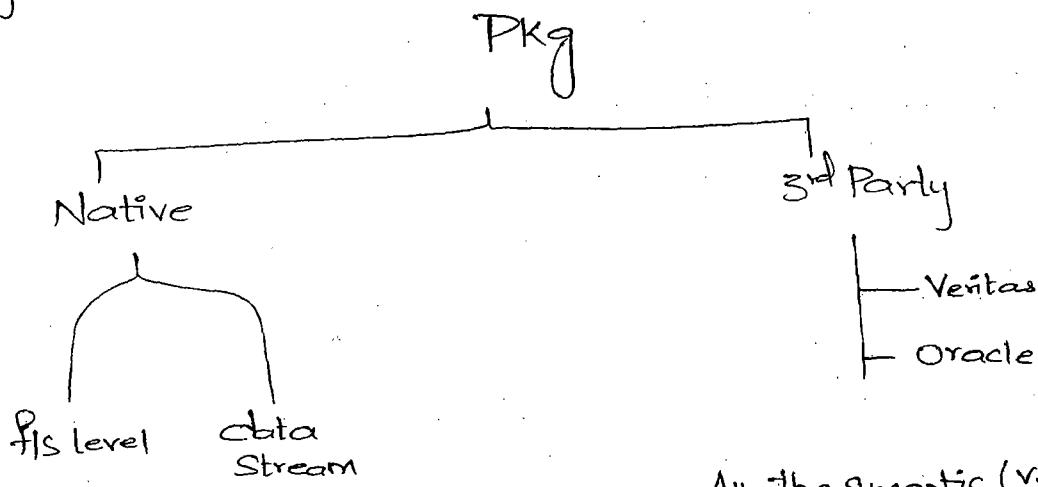
Eg.: SUNWman

Use of Package:- In Order to access a Service which not comes along with Operating System. Then we go for Packages.

- Generally all additional Packages are located under /lib Cd.

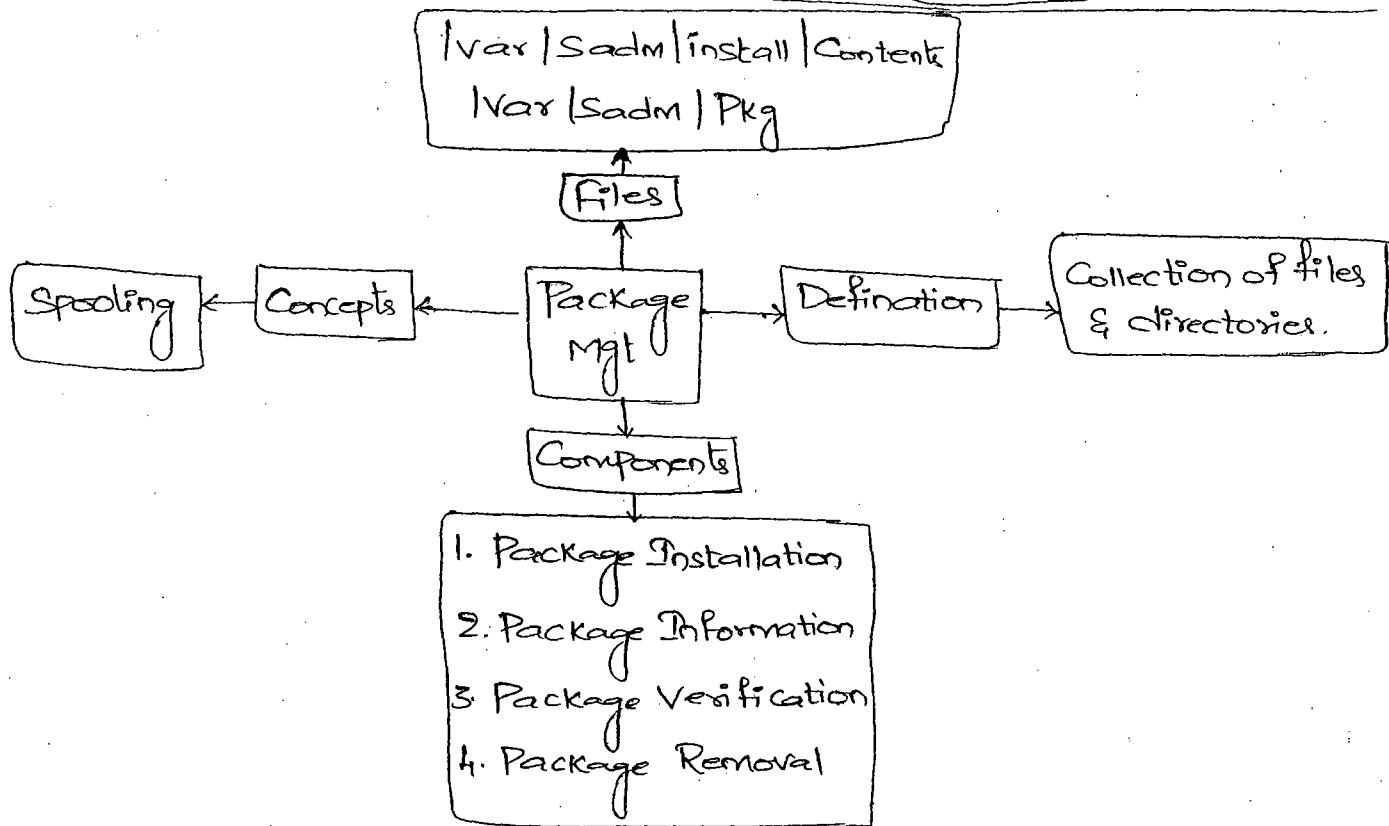
- Package Mgt. / Package Admin. enables you to adding (a) Removing the Packages.
- O/S also Collection of Cluster related Packages.
- Each & Every System files are also Emerged in a Pkg
- Even Cmds & daemons are also having Pkgs
- All the Basic Cmds Main Package is SUNWCSU

Types of Packages:



All the Symantic (Veritas) Pkg start with

VRTS xx



- If any command is not working properly there is a last choice to check the package consistency.

Native Packages:

1. fileSystem level:

Generally these type Pkgs. found in directory format.

2. Data Stream: All the data stream Pkgs. are ^{found} in file format, which are having 2 or more binary executable files.

- When we download any Pkg from Internet then that

Particular Package extension is ".Pkg".

- all datastream Packages end with ".Pkg" extension

- The need of datastream is that all the Pkg in fls are in the form of dir and we cannot download & share the dir, so we transfer the fls to data stream Pkg.

Commands:

Pkgadd (Installing)

Pkginfo (Info.)

Pkgchk (Checking)

Pkgrm (Removing)

Pkgtrans (Converting dir format - file format).

Mount Cd-rom automatically.

```
# cd /etc/init.d
```

```
# ./volmgt Stop
```

```
# ./volmgt Start
```

```
# df -h
```

- The daemon for cd-rom devices is **"void"**.

[#Pgrep void] ↴ (Used to check daemon is running (or) not).

- How to mount Cd-rom Manually?

```
# devfsadm -c cdrom ↴
```

```
# iostat -en ↴
```

(or)

```
# cfgadm -al ↴
```

```
# mkdir /cd ↴
```

```
# Mount -f hsfs /dev/dsk/Ct0d0s0 /cd ↴
```

```
# df -h ↴
```

- How to add / install a Package:

```
# cd /cd/Cdrom0/Solaris-10/Product ↴
```

```
# Pkgadd -d . SUNWman ↴
```

do you want to add : Y

(or)

```
# Yes | Pkgadd -d . SUNWman SUNWm... ↴
```

When a package is added no. of location gets updated

```
#vi /var|Sadm|install|Contents
```

- We can find those locations at ↑

```
# Pkginfo | grep <Packagename>
```

- lists whether package is added or not.

```
# Pkginfo -l <Package>
```

- gives entire info. of Pkg.

```
# Pkgchk <SunWman>
```

- to check consistency.

```
# Pkgrm <Packagename>
```

- to remove package.

```
# Yes | Pkgrm <Packagename>
```

(copying)
Spooling or Package:

- The default directory for spool

/var|spool|Pkg

Enter into cdrom Mountpoint

```
# Pkgadd -d . -s Spool <Packagename>
```

Specify
Spool dir.

Customize:

```
# mkdir lspool
```

```
# Pkgadd -d . -s lspool <Pkgnames>
```

How to transfer filesystem level Package to data stream Pkg?

- Before transferring Spool that Pkg. into | Var|Spool|Pkg.

tfile <filename> - displays what type of file it is.

Pkgtxns -d . | var|spool|pkg

(datastream)
SUNwoman.Pkg

SUNwoman ←
(fbs)

which ls } ↳ : displays the absolute path.

Pkgchk -l -P /usr/bin/ls }

Demo for Pkg Admin:

Insert 4/4 cd

t devsdadm -c cdrom

t iostat -en (in x86)
(∞)

Cfgadm -al

cd /etc/init.d

- /volmgt stop
- /volmgt start

Pgrep void

Mounting

Pkill -9 void

Pgrep void

Cfgadm -al

#mkdir |cd

Mount -f hsfs /dev/dsk/C0t2d0s0 |cd

#df -h

Install Pkg:-

cd /cdrom/cdrom0/solaris_10/Product

ls | grep -i SUNWman

* Yes | Pkgadd -d .SUNWman

Pkginfo |grep SUNWman

Pkg Consistency:-

Which telnet ↴

1usr/bin/telnet

Pkgchk -l -P /usr/bin/telnet

Pkgchk SUNWtelnet ↴

#Pkginfo -P <Pkgnname>: used to list Partially installed Pkg.

more /var/sadm/install/Content

Converting file format to datastream format:

Pkgadd -d . -s spool SUNWman

cd /var/spool/PKG

file *

SUNWman directory

Pkgtrans /var/spool/PKG SUNWman.Pkg SUNWman

Password Breaking:

Stop + a

Put 1st cd.

OK > boot cdrom -s

Single User mode.

(all critical filesystem)

OK>boot net -s (without inserting cd)

in x86
Put 1st cd
type of inst. boot -s

df -h

In x86. keep 1st cd
init 6 &
b -s &

pwd

/tmp/root (in single user)

mkdir ram

Mount /dev/dsk/c0t0d0s0 /tmp/root/ram

df -h

cd ram

ls

vi etc/shadow

delete root Password by Pressing: x.

:wq

Reboot.

'OK' Boot Prompt:

O.b.P

Stop + a.

okbanner — Shows you System info. (OS release date, Processor info, Speed of Processor. Open No:

OK> • Speed ↴ : Show Speed of CPU
UPA
PCI

OK> • Version ↴ : Show Version of OBP &
POST

OK> Page ↴ : Clear the Page.

OK> • net-addr ↴ : Show your ethernet address.

OK> Printenv ↴ : Show your Environment Variables.

Var. name	Value	default value.
--------------	-------	-------------------

'q' - Exit from Particular Cmd.

To Change.

OK> Setenv auto-boot = false.

OK> Setenv boot-device cdrom disk net

OK> Set-defaults ↴ - Set the default Values.

OK> Deralias ↴ : Show alias name with Physical Path.

OK> Invalias cdrom Raj ↴

OK> Invalias Raj

OK> Show-disks ↴ : Physical Path of disks.

OK> Show-tapes : Physical Path of tapes

OK> Show-devs : Physical Path of devices.

OK> Probe-ide : Ide disks info.

OK> Probe-scsi : Scsi disks info.

Booting Commands:

OK> boot

OK> boot cdrom (boot from cdrom)

OK> boot cdrom -s (Single userMode)

OK> boot -a (Interactive booting)

OK> boot -v (verbose)

OK> boot -x (Recursively booting).

OK> date

OK> Power-off (Shutdown your system)

OK> go (to os level)

5/10/09

PATCH Mgt:- (Service Pack in x86)

: Patch is a Collection of two (Or) More Packages

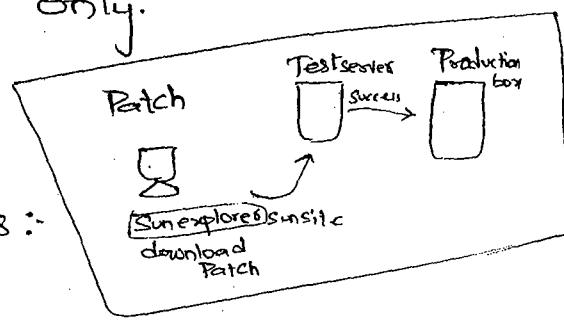
Which will Replace the existing files & directories.

- When we Need to Install a Patch?

1. For Proper Execution of SW
2. In order to upgrade Os one Version to another Version.

Before Installing a Patch it's Mandatory to take a backup of all Main Configuration files and some Commands output also.

Note: Always we need to implement Patches in Mainta.
Hence Mode only.



df -h
lctl system
lctl vlistab

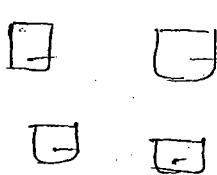
Types of Patches:

1) Standard:

Generally These type of Patches are Used to

trouble shoot errors like Sun software, hardware.

2) Recommended: (Security Patches)



When there is same error on multiple systems, then Security Patches are come into Picture.

3. Firmware:

Generally these type of Patches are used to Upgrade the firmware (O.B.P)

4. Patch Cluster:

Combination of Standard, Recommended & Firmware Patches.

We can find out Patch clusters generally in archive file format.

Commands Related to Patch:

Patchadd (add a Patch)

Patchrm (Removing a Patch)

Showver (Used to display currently installed Patch)

Location for Patch # cd /var/sadm/Patch

Patch Naming Convention:

10520.01
~~~~~ ~~~~~  
base name version

# Patchadd <Patchid>

# Patchadd -P ] ↳ Used to display currently installed Patch  
(or)

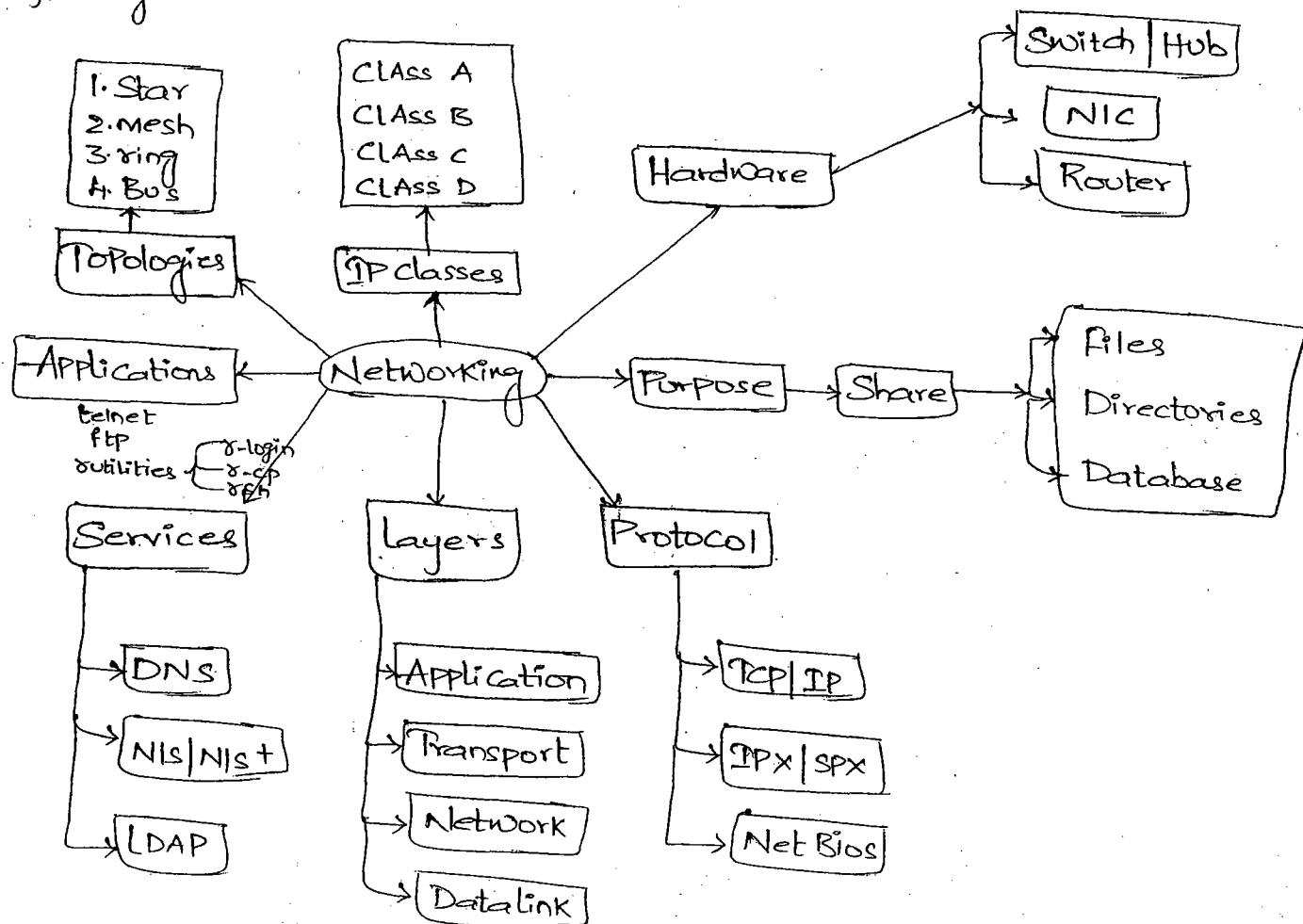
# Showver

# NETWORK CONCEPTS:

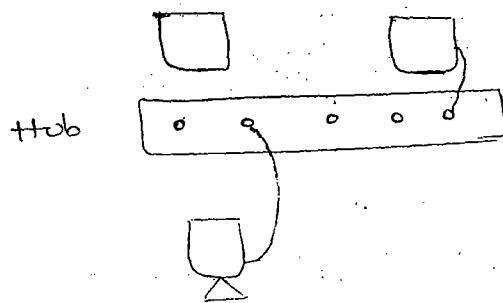
Network: The interconnection b/w 2 or More Systems or a group of Systems is called Network.

Networking: The communication b/w these interconnected devices is Networking.

Basically used for Share the Resources as well as filesystems.



## Switch / Hub



Hub: Each and Every time a System Sends a data to another one it will check each and every Port of the destination Systems.

Switch: Once the Connection established it keep on transferring data.

Router: Pick the data Packets and send through the n/w that is to be sent.

Protocol: It is a set of Rules & Regulations which is used to communicate two or More Systems.

TCP/IP: Can Support Wired n/w and it Provide acknowledgement for the System.

to know the Packets 

|       |                       |        |
|-------|-----------------------|--------|
| #Ping | < Remote machine IP > | : usec |
|-------|-----------------------|--------|

to list n/w Packet info.

UDP: Support Wireless. It won't Provide acknowledgement for the System.

## IP classes:

Class - a : 1 - 126 N H H H (Can connect n no. of sys.)

Class - b : 128 - 191 N N H H

Class - c : 192 - 223 N N N H

based on Subnet Mask.

## TCP/IP layers:

Application

Presentation

Session

transport

Network

datalink

Physical

(tcp/ip, UDP)

(addressing) the frames.

(bits - frames)

(bits) Maintain Physical connectivity.

Application Layer: Responsible for Provide apps which is used to communicate the Systems.

## Configuring NIC Card:

New Card:

# dnf install net

# cat /etc/path-to-inst | grep net

To know the Nic Card installed & name also.

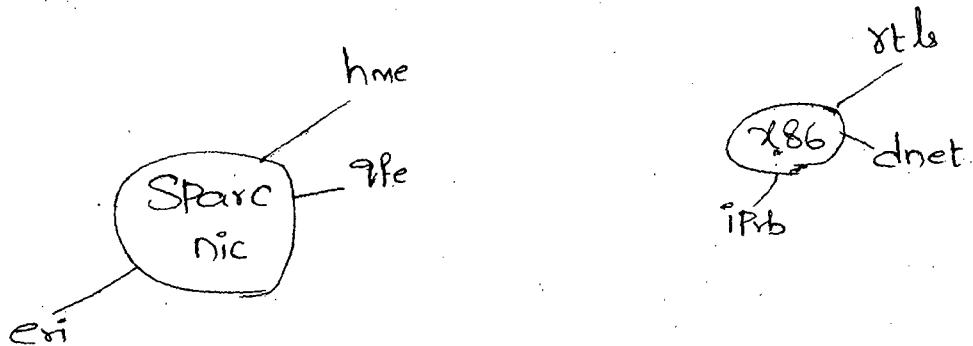
e.g. Pcm@abc..... net "hme"

# ping -v | grep net

In "sparc" manually.

In x86 we need to open the file manually to see nic card

info # Vi /etc/Path-to-inst ↴



# ifConfig -a ↴ used to display the nic card info.

How to enable nic card?

#ifConfig hme0 plumb ↴

How to disable

#ifConfig hme0 unplumb ↴

How to Provide Primary IP Address for Nic Card.

# ifConfig hme0 200.200.0.1 UP ↴  
nic card name      IP  
Key words: up (Participate in net)  
down (cannot Participate)

to Remove Primary IP address

#ifConfig hme0 unplumb ↴  
0.0.0.0

## Creating Virtual IP Address :-

addif → Keyword used to Provide Virtual IP.

```
#ifConfig hme0 addif 200.200.0.2 UP
```

(or)

```
#ifConfig hme0:1 Plumb
```

```
#ifConfig hme0:1 <IP> UP
```

How to Remove Virtual IP ?

removeif → Keyword to Remove Virtual IP.

```
#ifConfig hme0 removeif 200.200.0.2
```

Files Related to IP-Address & Host Name: (alias name for IP address).

```
# hostname wstsun09
```

Used to Create hostname.

```
# hostname
```

display hostname.

```
# vi /etc/hosts
```

→ Resolving IP-host  
host-IP

200.200.0.1 wstsun09

(IP) (hn.)

```
# vi /etc/linet/linodes
```

(at system bootup time System will Read &  
Assign IP & host)

200.200.0.1 wstsun09

`#vi /etc/hostname` ↴ (Responsible for maintaining only host name).

`#vi /etc/hostname. <interface>` ↴ (This is the Main Configuration file for IP address & HostName).  
hostsun9

`# vi /etc/inet/hosts` ↴  
it is linked to `/etc/hosts` (soft link file).

If we want to save Primary IP address Permanent.

`/etc/hostname.hme0` ↴

Virtual IP Permanent

`/etc/hostname.hme0.vi` ↴

Telnet: Telnet is a binary Prog. Used to Communicate one System to another System.

Default Port no. for Telnet : 23

All the default Port no. are located under

logical end of service, it is responsible for maintaining System calls

System Calls

- enable
- accept
- read
- write

`/etc/services`.

Daemon for Telnet is `in.telnetd`

To see Telnet running (or) not

`Svcs -a | grep telnet` ↴

wstsun1 200.200.0.1

#arp -a ↴ display

Currently Available Systems  
in n/w.

#rwp ↴ Remotely upend

running systems.

wstsun1  
200.200.0.1



Before log into the system  
we have to check Ping  
Status.

#Ping 200.200.0.2 ↴

up; alive ↴

Now let us try to login into  
System using telnet cmd.

#telnet 200.200.0.2 ↴

it will ask

Login: root ↴

Password: xxxx ↴

for the first time we get an  
error like not on System  
Console then

wstsun2  
200.200.0.2



#vi /etc/default/login ↴

18th .

#CONSOLE = /dev/console

Now again login

# Csh ↳ To know on which

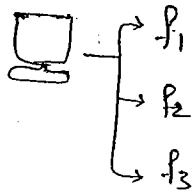
## FTP: (file transfer Protocol)

We can transfer only files from one system to another

Port no; for FTP : 21

Daemon is in.ftpds

200.200.0.1



200.200.0.2



# ftp 200.200.0.2 ↳

Login: root ↳

Pass: xxxx

Get's error in first login as  
"Login failed". but gives  
FTP Prompt.

FTP> bye ↳

Login: root

Password: xxxx ↳

FTP> ls

vi /etc/ftpd/ftpusers ↳

→ this file is responsible for  
restrict the ftp access.

#root

In remote machine  
Comment the entry (or)  
delete.

## K-Utilities:

rlogin (remote login)  
 rcp (remote copy)  
 rsh (remote shell)

- Don't have default  
Port no:

rlogin: is a command used to login Remote Machines without asking the Password. if the remote Machine is trusted.

Trusted relationship.

#Cat > / .rhosts ↴ Main Config. file for Entire r-utilities.

+

Ctrl+d ↴

- Create this file in Remotemachine.  
it will become trusted.

Once We Create this file <sup>with +</sup>, anyone can login.  
for only one Particular ~~Machine~~ Machine.

Cat > / .rhosts

200.200.0.4 :root

Ctrl+d ↴

rcp :- (remote Copy)

# rcp lf1 200.200.0.2 : / di

# rcp 200.200.0.2 : / f22 . ↴

# rcp -r ld22 200.200.0.2 : /

## Rsh : (Remote Shell)

Using Rsh we can access Remote Machine Console.

```
# Rsh 200.200.0.2 mkdir ldee ↴
```

Note: try to Create a file in Remote Machine with the help of Rsh.

```
#Rsh 200.200.0.1 'echo "this is test file"' ↴/p1<
```

### Demo:

```
arp -a ↴
```

```
* Rup ↴
```

```
# hostname ↴
```

```
# Ping 200.200.0.14 ↴
```

```
Ping -s ↴
```

```
# telnet 200.200.0.14.
```

not on System Console

~~18~~ line Comment ↴

Login

```
# Csh
```

```
Wstsun14>
```

```
# Cat /etc/services
```

### FTP Demo:

Wstsun80

```
FTP 200.200.0.14 ↴
```

Login:

Password:

Login failed.

Wstsun14

```
vi /etc/ftpdl/ftpusers
```

Comment or Uncomment root entry.

Save & quit

90

ftp> bye <

#Ptp 200.200.0.14

Login:

Password:

ftp> !clear (to clear screen)

ftp> ls < (displays remote machine files.)

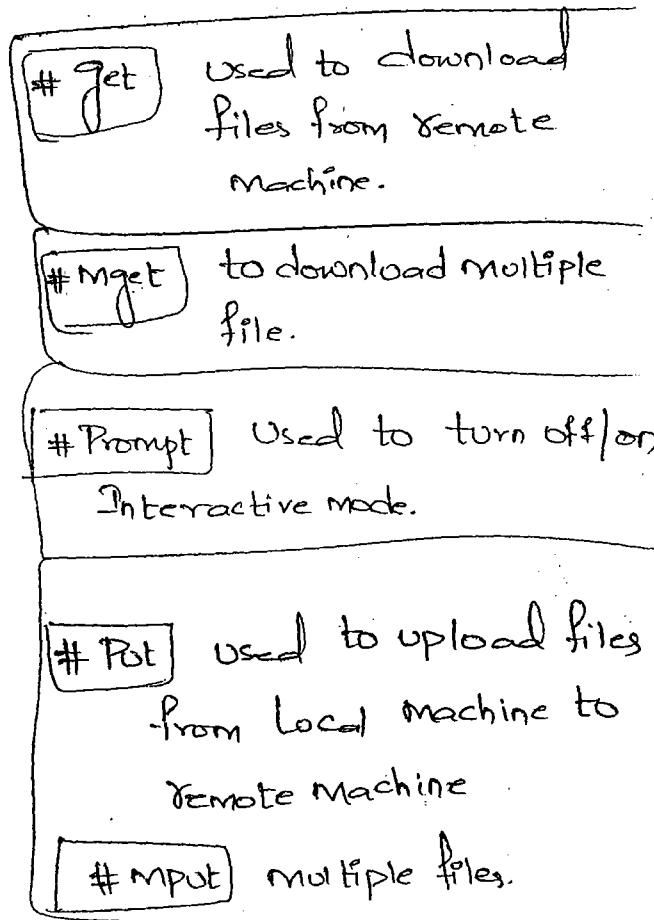
ftp> !ls < (displays local machine files)

ftp> get fi <

ftp> mget f1 f2 f3 <

ftp> Prompt

ftp> Put file1



ftp> hash < (mark)

ftp> ? (used to display Possible Cmds under ftp Prompt).

ftp> Close (used to close session temporarily)

ftp> open (used to open session).

ftp> bye (Used to come out from the Particular Session)

## X-login demo:

wostsun14 # cat > /xhosts

+

Ctrl+d.

wostsun80 # xlogin 200.200.0.14

# Csh.

wostsun14 #

## XCP:

wostsun14 # mkdir /xdir ↴

wostsun80 # xcp -x 200.200.0.14:/xdir . ↴

## Xsh:

wostsun80 # xsh 200.200.0.14 mkdir /sun ↴

---

## Nic Card Config Demo:

# Rtdiag -v | grep -i net

# Cat /etc/path-to-inst | grep -i net

# arch

in x86.

# vi /etc/path-to\_inst

# ifconfig -a

```
# ifconfig eth0 Plumb
```

```
# ifconfig -a
```

```
# ifconfig eth0 200.200.0.80 up.
```

```
# ifconfig -a.
```

### Virtual IP:

```
# ifconfig eth0 addif 200.200.0.157 up
```

```
# ifconfig -a.
```

```
# ifconfig eth0 delremoveif 200.200.0.157 up
```

```
cat /etc/hosts
```

```
/etc/inet/ippnodes
```

## JANA

Internet Assigned naming Authority.

↓  
Provides IP { static → standard  
dynamic }  
to identify systems intra/internet → it may change  
"dhcp"

Type of IP class = class - a, b, c, d, e

A →  $\boxed{1 - 126}$ , IPV4 = 4 octets

$\boxed{N} \cdot \boxed{H} \cdot \boxed{H} \cdot \boxed{H} \rightarrow$  million no host  
1-126 0.0.1

(255) 255.0.0 → subnet mask

B → 128 - 191

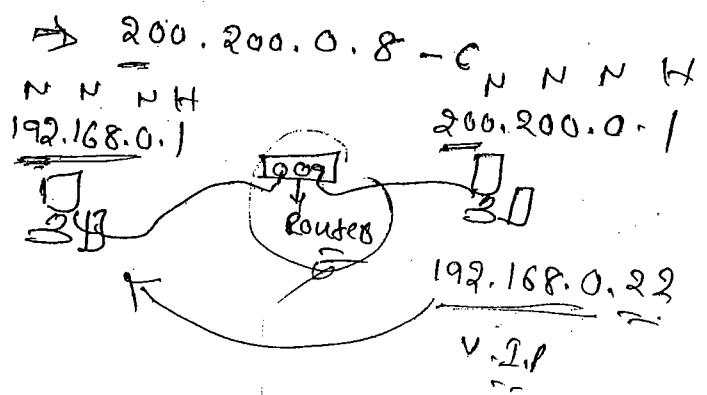
$\boxed{N} \cdot \boxed{N} \cdot \boxed{H} \cdot \boxed{H} \rightarrow$

255.255.0.0

C → 192 - 233

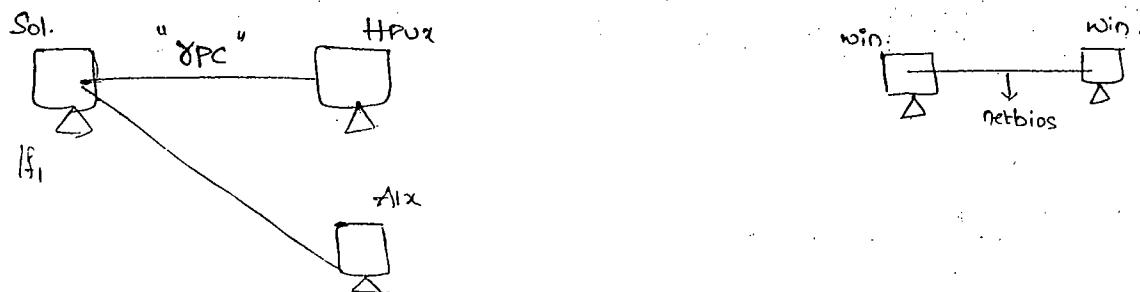
$\boxed{N} \cdot \boxed{N} \cdot \boxed{N} \cdot \boxed{H}$

$D, E \rightarrow R \neq$



91.2.9.78

# NFS (Network file System).

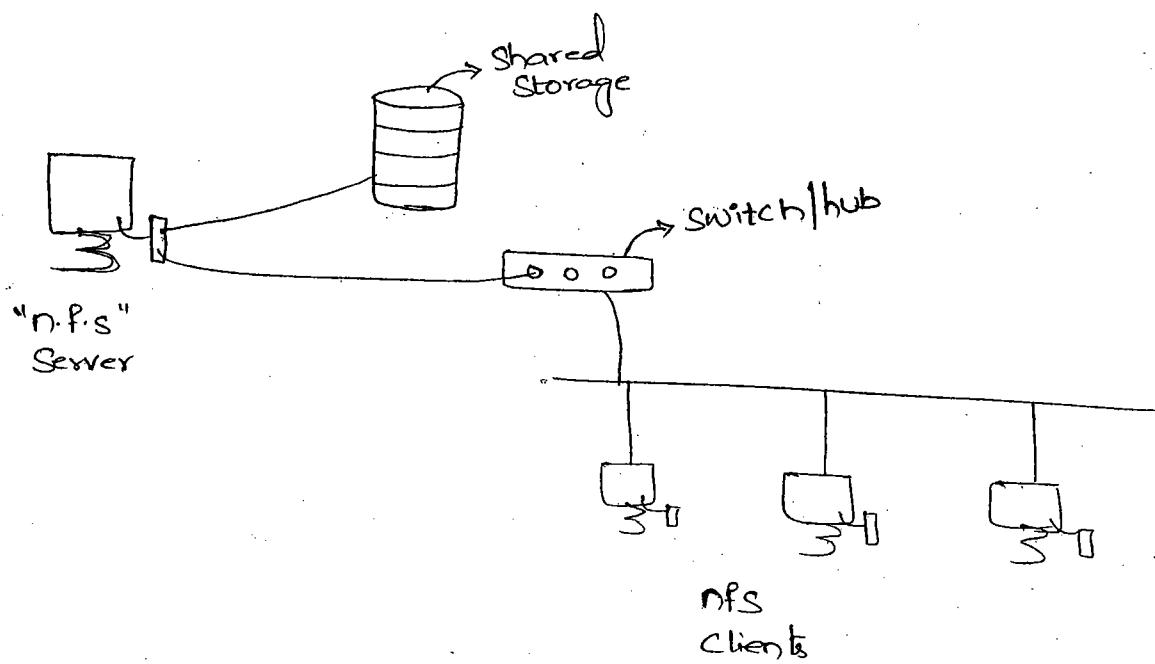


- NFS is a special type of filesystem developed by Sun Microsystems.

- NFS is used to share files & dir. and databases over the n/w.

- Port No: for NFS is : 2049.

- We can centralize users db's & applications.



- NFS uses R.P.C Protocol which is used to share files & dir

- NFS can support any Unix platform Shared Resources.

Versions in NFS.

NFS 2

NFS 3

NFS 4 → default in Solaris 10.

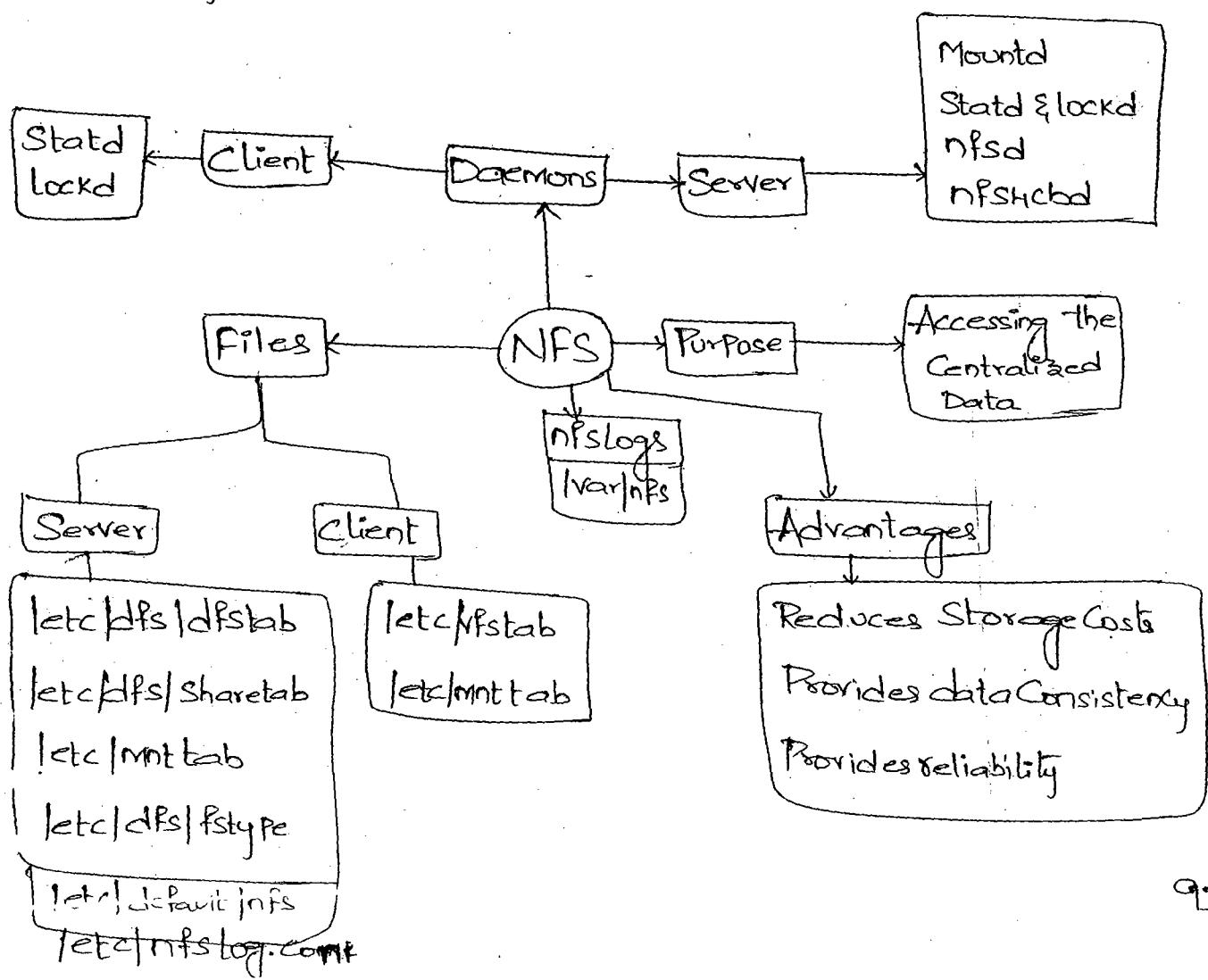
### Purpose of NFS:

- Saving wastage of resources.

- Easy to administrate.

- Providing common storage for all Client Machines.

- Availability.



Eg.: Sewer

## Create file & Mount it

S5 → |users.

# Share | Users |

## Client

```
# Showmount -e <remote Server IP>
```

200·200·0·8

OLP: \Users

```
#mkdir test
```

```
#mount -F nfs 200.200.0.8:/Users
```

1test ↴

files :

## 1. Server Side files:

/etc/dfs/dfstab:

etc|dfs|dfstab: - This file is responsible for making the shared resources permanent.

- It is similar to /etc/vfstab.

- At the System Startup time it tries to read this file & Share the attributes automatically.

#vi /etc/dfs/dfstab

etc/vfstab <

Share -F nfs /my nfs

Mountall ↴

Unmount all ↴

: wq ل

#Shareall ↴ — codes dfstab file & Share all the attb.

#ShareAll! — Xcode

#unshareall < —

## /etc/dfs/fstypes :

- Responsible for Maintain all NFS related  
File System.

### # vi /etc/dfs/fstypes

nfs → 1<sup>st</sup> Pref  
autofs → 2<sup>nd</sup> Pref

#Share /my nfs

when we share /my nfs by default  
as 1<sup>st</sup> Pref. it takes nfs.

## → /etc/default/dfs

Local = UFS

#mount ss tuser

when we mount ss it will  
automatically takes UFS file system  
Otherwise if it is cdrom we give  
-F hsfs

## /etc/dfs/sharetab :

- Responsible for Maintain Shared Resources

which are Shared in Server Side.

- System will take care of this file.

#share /my nfs.

- maintain currently  
Shared information

mount ss tuser.

/etc/mnttab : maintain all  
mount info.

## /etc/default/nfs :

- It is Main Configuration file for NFS

Responsible for maintaining NFS Version.

- If we want to change NFS Version. Manual  
We need to edit this file.

```
# Vi /etc/default/nfs
```

```
# Client - VERS_min = 2
```

```
# Client - VERS_max = 4
```

After modifying /etc/default/nfs file we need to restart the service. Then only changes will effect.

```
#svcadm restart nfs/server
```

/etc/nfs/nfslog.conf :

Responsible for maintaining logs Configuration information.

/etc/mnttab & /etc/vfstab are also default in server.

Client Side files :

/etc/mnttab : maintain all mount points

/etc/vfstab : to make fts permanent

/var/nfs : log msgs.

/etc/default/nfs :

After modifying /etc/default/nfs file needs to restart.

## Commands Related to NFS:-

### Serverside:

|            |            |                                         |
|------------|------------|-----------------------------------------|
| Share      | Mount      | dfmount's (works only on nfs version:3) |
| Shareall   | Mountall   | dfshares                                |
| Unshare    | Unmount    | nfsstat                                 |
| Unshareall | Unmountall |                                         |

# dfmount's : (works only on nfs Version:3)

Responsible for giving O/P of nfs clients.  
(How many)

The O/P comes from file **etc/lmntab**

Using **#fsstat**  
(in Version 4) we can get how many shared and how many users.

# dfshares : Show Shared folders

Based on **etc/dfs/sharetab** file we will get O/P

# nfs stat : How many read, write O/Ps going on. List nfs statistics.

# nfs stat -m : displays version of nfs.

### Client Side:-

Showmount

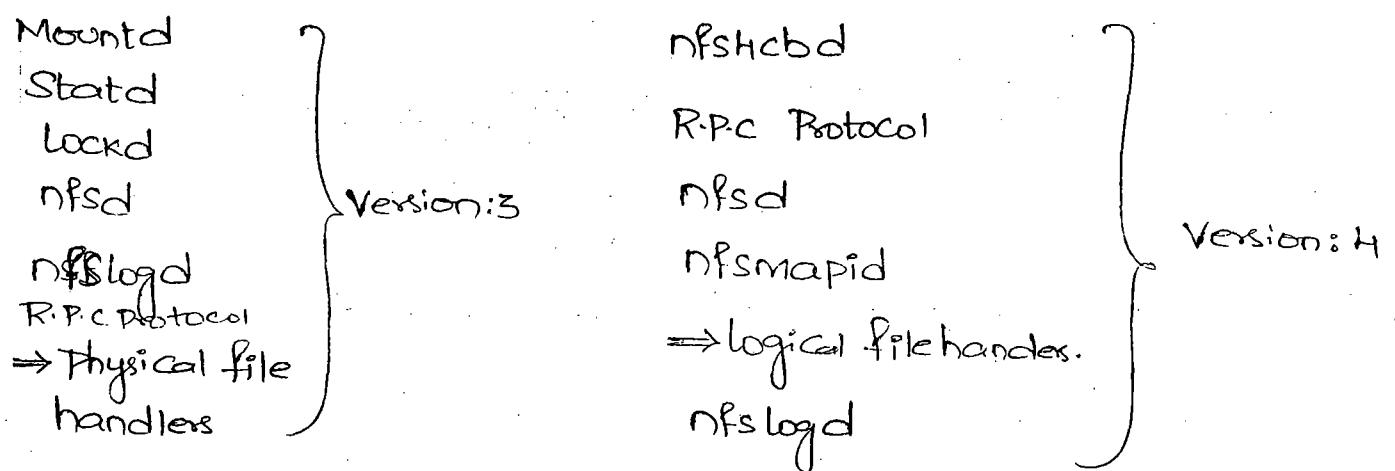
Mount

Mountall

Unmount

Unmountall

## Daemons :-



Nfsd: Responsible for responding the Client Request for Mounting any fIs.

- handles client request
- located at Serverside.

Mountd : Responsible for providing the mount request to the nfs server.

Stated : Responsible for maintain the status of shared resources from server side.  
- It is responsible for check heartbeat in both Client & Server.

lockd : Responsible for locking the remotely mounted filesystems on the client side.  
- When ever server is not responding this daemon will locks the resources.  
- This entire operation will take care by R.P.C (Remote Procedure Call) Protocol.

## NFSMapd:

- New feature in Sol. 10.

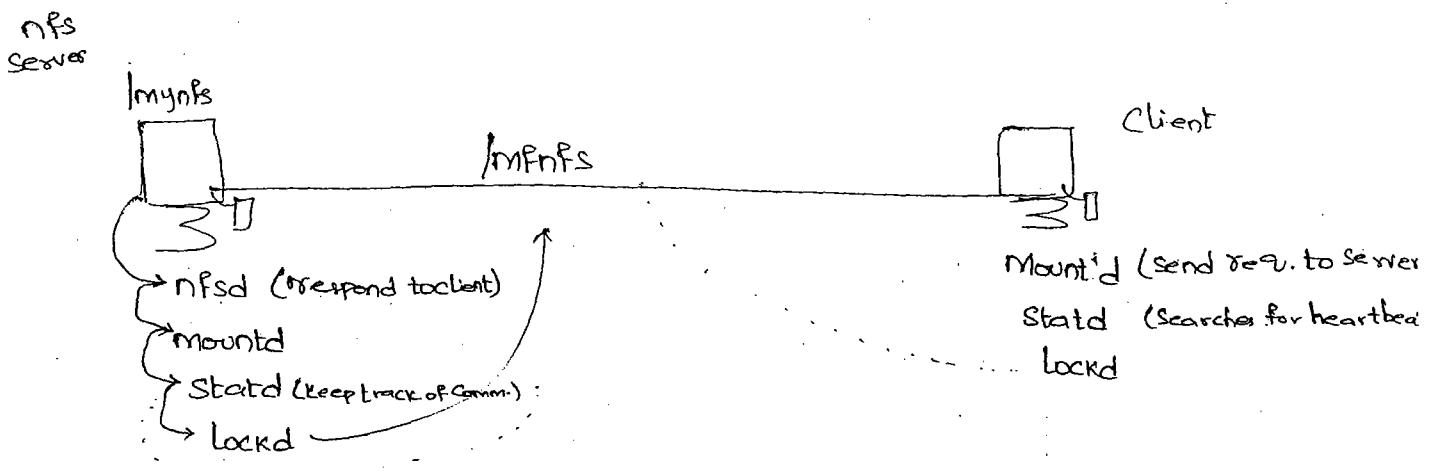
- Maintain mapping info. on Client & Server Side.

## NFSHcbd : (Client back deamon)

- Responsible for maintain the end-end comm.

b/w Server & Client.

## NFSlogd: Responsible for generating the log messages.



Demo: nfsServer : 14      nfsClient : 80

wstsun80

### Server side Configuration

Create a f/s & mount it.

- After mounting.

- Change Permissions.

- # chmod 777 /my nfs

- How to share a f/s in cmdline

# Share -F nfs /my nfs

### Client side Config.

# Showmount -e 200.200.0.14

# mkdir /testnfs

# Mount -F nfs 200.200.0.14:/my nfs /testnfs

# df -h

# Share ↴

How to Share files in filelevel.

vi /etc/dfs/dfstab

Share -f nfs /my nfs ↴

:wq ↴

# shareall ↴

Share -f nfs -o ro = 200.200.0.1 /my nfs ↴

options. read only.

(Give Perm. to only one client  
that too read only Perm.)

# umount testnfs

# vi /etc/default/nfs

goto line no: 61

Uncomment it.

make 4 to 3

:wq ↴

# unshareall

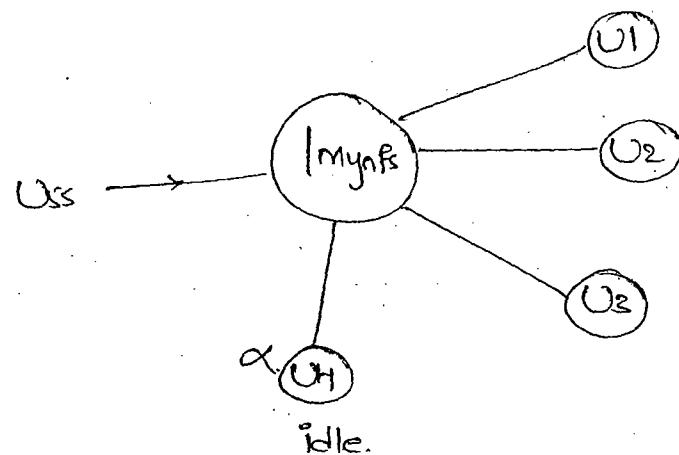
# svcadm restart nfs/server

# svcs -a | grep nfs/server

online

# lfmounts (etc/mtab)

- Autofs: is a special type of ffs. which is used to Mount (or) Unmount a share point on Client side.
  - generally this type of ffs is used for reducing the network traffic.



Two types:

direct method

Indirect method. (nis)

Direct Method:

|                                    |   |                    |
|------------------------------------|---|--------------------|
| /etc/auto-master<br>/etc/auto-home | { | Main Config. files |
|------------------------------------|---|--------------------|

Command:

"#automount"

"automountd" — deamon

- Autofs is client side ffs.

## Server

Create f/s & Mount it

```
# Share -f nfs /myfs.
```

## Client

```
# showmount -e 200.200.0.14
```

/myfs.

```
# mkdir -p d1/d2/d3/14
```

```
# vi /etc/auto-Master
```

|  
|- auto-home mapping file.

```
# vi /etc/auto-home
```

```
d1/d2/d3 200.200.0.14:/myfs
```

```
:wq
```

Here it will be mounted in d3.

```
# automount -v
```

↓  
Automountd

auto-Master      auto-home

```
# cd d1/d2/d3
```

```
# df -h
```

## DEMO:

wstsun14

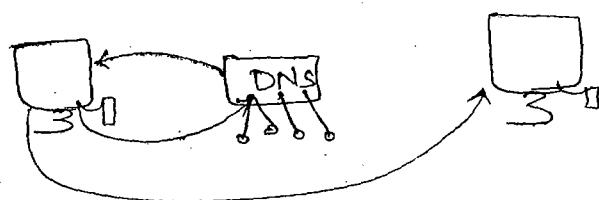
Create f/s & Share it

## Client

19/10/01

DNS: - is a domain naming system.

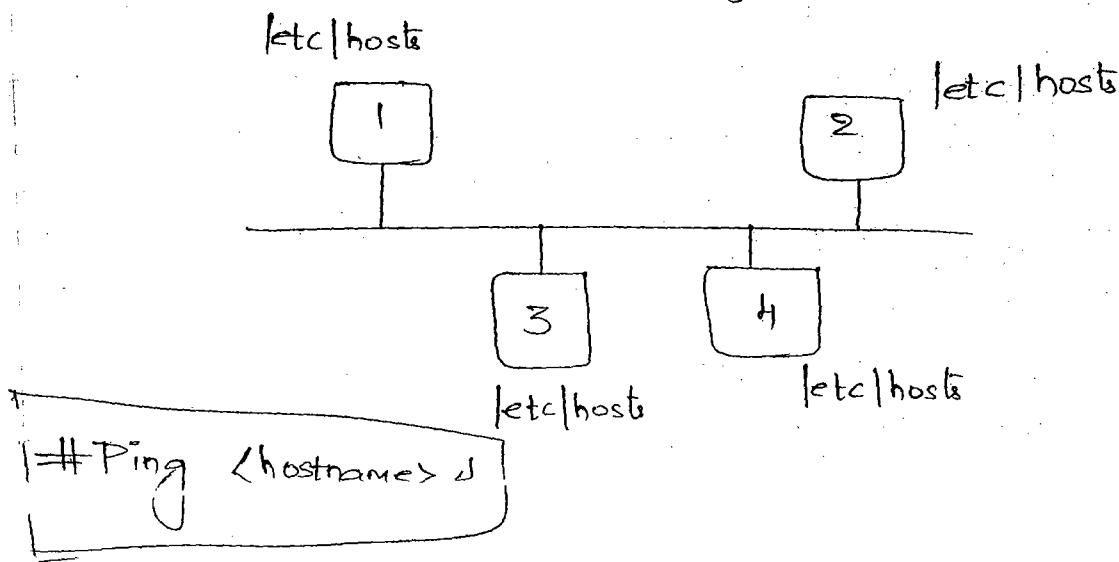
Domain: is a collection of two (or) more systems which are participating in network.



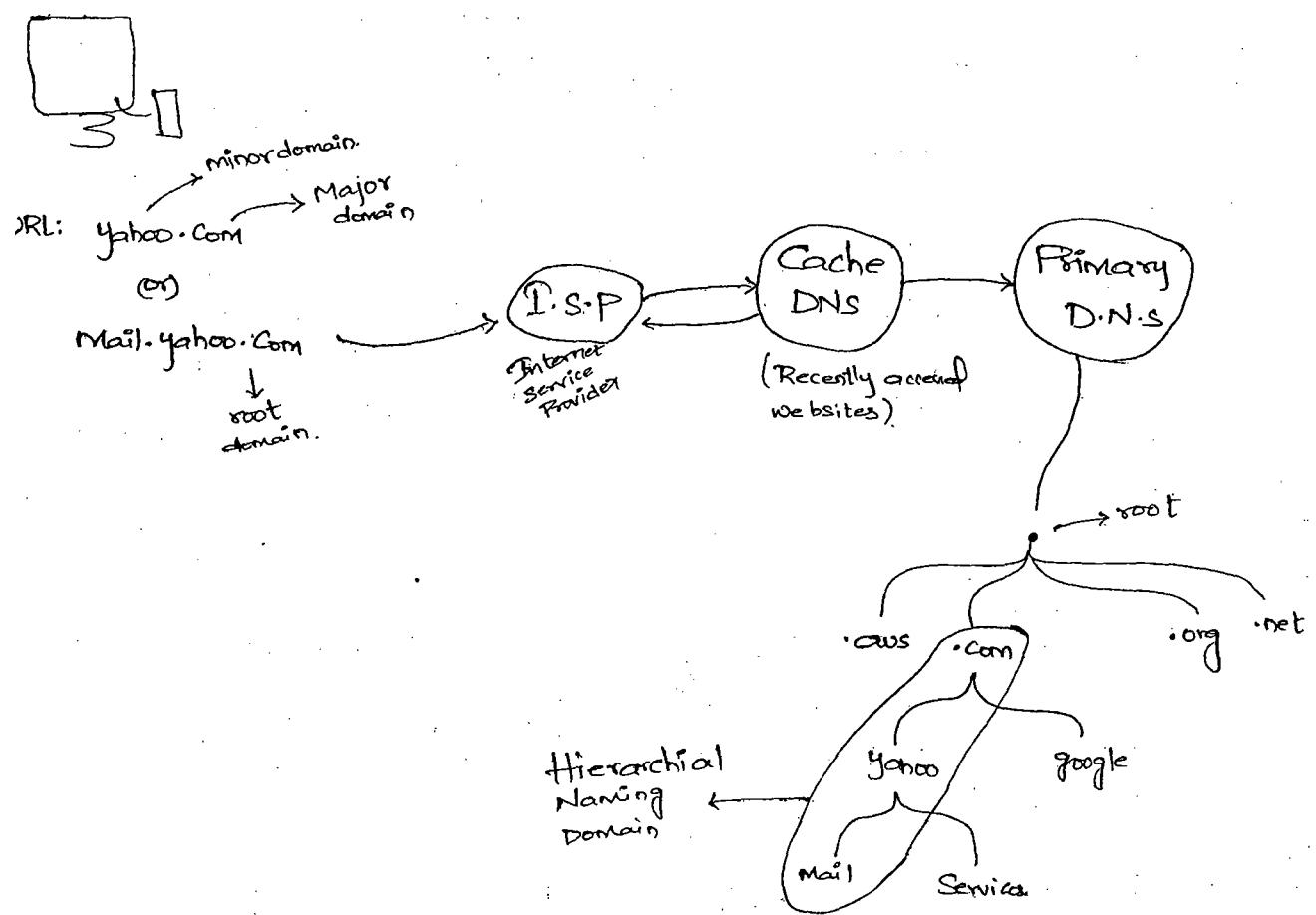
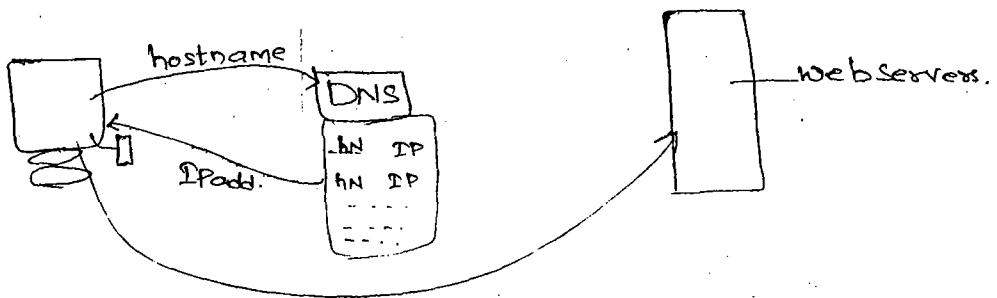
- Domain Naming System is a distributed db system.  
basically used for resolving the IP address to host name. hostname - IP address.

- Generally **[etc/hosts]** file is responsible for resolving the IP address - hostname & hostname - IP address.

- In Entire Environment each & every system can comm. with hostnames only not with the IP address.



# How DNS Works?



## Types of Servers:

1. Root D.N.S
2. Primary DNS
3. Secondary DNS
4. Cache DNS
5. forwarder Server

## Packages for DNS:

→ **SUNWbind**

→ **SUNWbindr**

**Port Number for D.N.S : 53**

**[etc]Services** → We can find all Port no.

**[etc]Services | grep domain.**  
53.

## Daemon for DNS

in.named → Sd.9

Named → Sd.10

### 1. Root D.N.S Server:

- Top Most level Server which is responsible for maintain the all Primary DNS Servers.

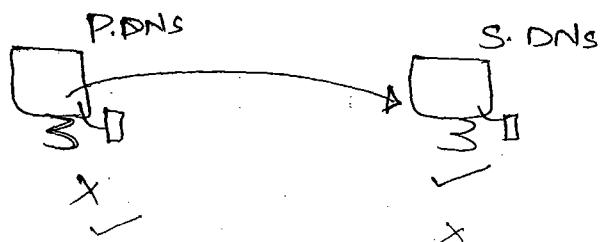
- Root Servers can be maintained by an organization called IANA (Internet Assigned numbering Authority).

### 2. Primary D.N.S Server:

responsible for  
- This Server is Maintain all db's info.

### 3. Secondary DNS Server:

- Can Call it as Slave Server.
- When the Primary DNS Server is goes down. Then this Server will act as Primary DNS.
- It regularly updates the db.info from Primary DNS Server with the help of "SOA" (Start of Authority) Records.
- SOA Maintains
  - Serial no:
  - Refresh
  - Retry
  - Expiry
  - T.T.L (time to live).
- The Secondary DNS Server used to provide load balancing.



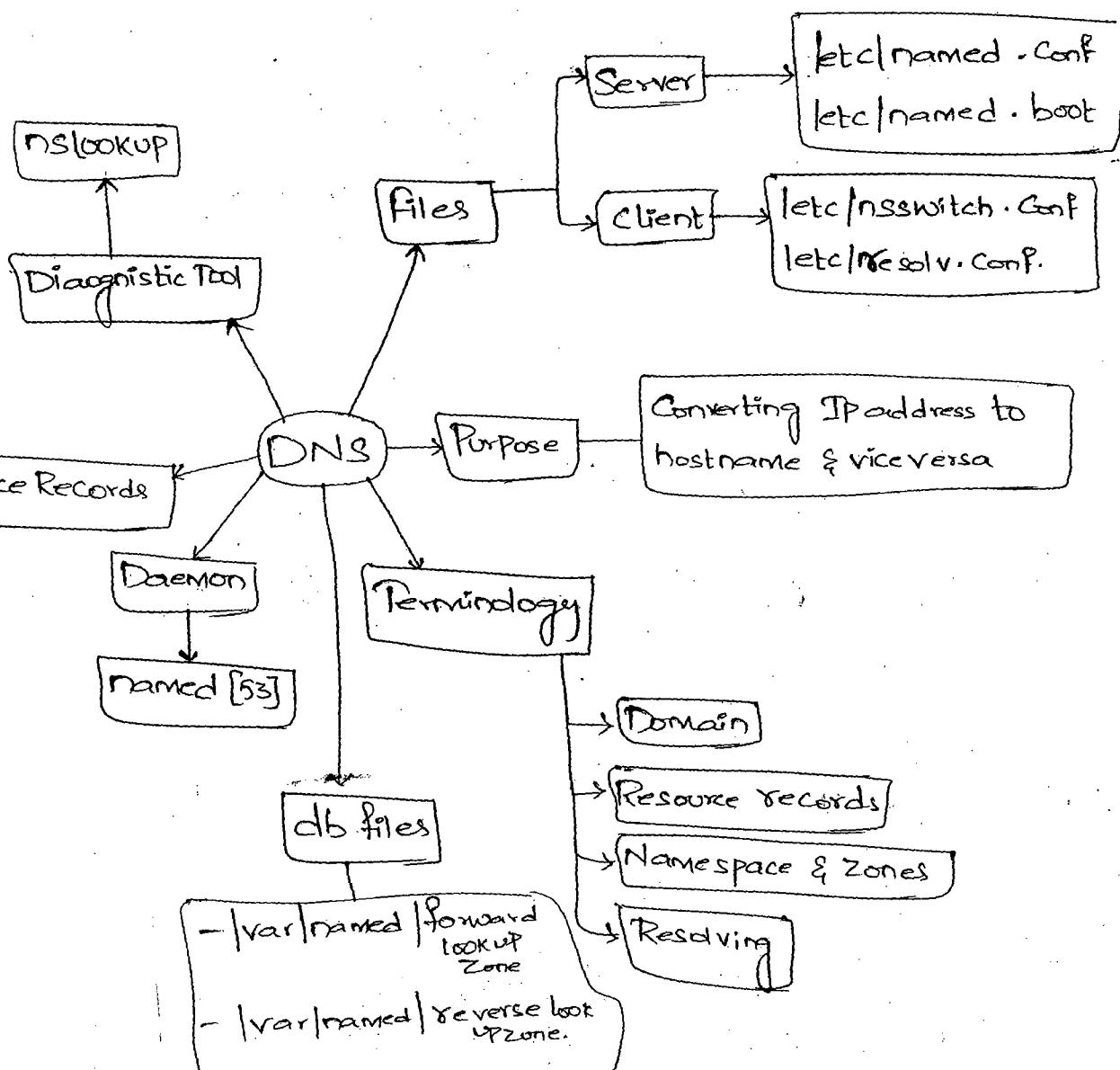
### 4. Cache DNS Server:

- It Maintain total db.info. which are recently Accessed from Primary DNS Server.

## 5. Forwarder Server:

- This Server is used to send the request made by the client to multiple DNS servers due to provide load balancing (or) load sharing.

- Forwarder DNS Server is logical one.



## Terminology:-

• Domain: logical Collection of Two (or) More nodes participating in Some network.

• Sub domains: large nw are divided into Sub parts. is called Sub domains.

Eg: yahoo.com → Mail. yahoo.com  
                                                Subdomain

• Resolving: is a Process of Mapping the hostname to IP address. & IP address to hostname by checking the 2 db files. which are located under

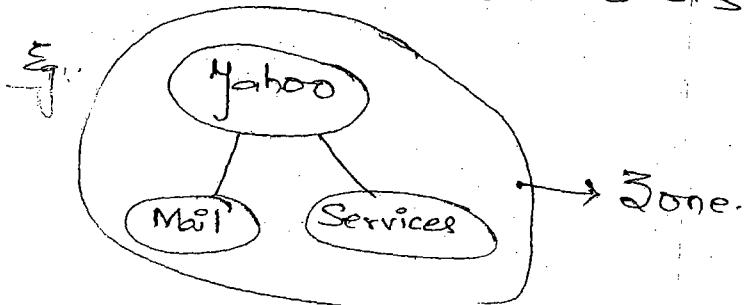
Var/named

h.name → IP → forward lookup zone

IP → hostname → reverse lookup zone.

ZONES: - It is a Collection of a hierarchical view of a Major domain.

- When the DNS Server gets updated, it is done on basis of Zone.

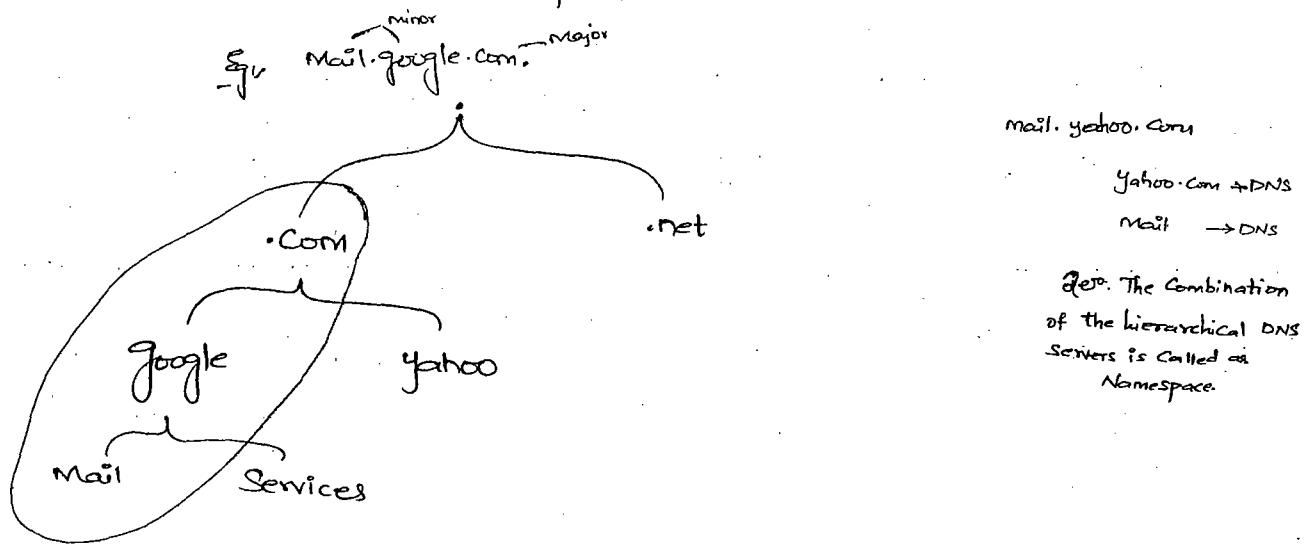


Q. 5. NameSpace: A System Which is Participating With  
the ~~single~~<sup>two or more</sup> DNS Server is Called Namespace.

- We have 2 types

\* Flat Namespace Eg. "westsun1" (Just name, & Collection of char).

\* Hierarchical Namespace.



Q. 6. Resource Records: - These are inbuilt Scripts which

are Provided for the Mapping Purpose.

- All Resource Records are Maintained by organization : BIND (Berkeley Internet naming distribution)  
&  
Arpa (American research Project authority).

Types:

SOA (Start of Authority)

IN (Internet)

NS (Name Service)

A (Address)

PTR (Pointer)

A: (Address)

- It Provides db info. for mapping Purpose.

SOA: (Start of Authority)

- This record identifies the Primary Name Server, Contact info, default Cache, TTL values, for all resource records in the domain.

NS: The Name Server record specifies the name server for domain.

PTR: The Pointer record specifies a hostname for an IP address used to resolving Purpose.

Data Base files:

Var/named

named.hosts (forward lookup zone)

named.rev (reverse lookup zone).

Configuration file in DNS Server:

etc/named.conf

Server Side

(contd.)

- This file is responsible for list out the where exactly the db files are located.

# named-checkconf

Used to check main configuration file.

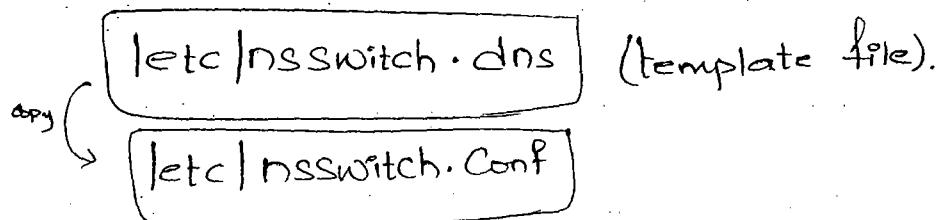
- It will check each & every line.

## Client side Main Configuration file

etc/resolv.conf

—responsible for Maintaining Name

Server & domain name.



# Vi /etc/nsswitch.conf

hosts: files dns

1<sup>st</sup> Priority — DNS.

### F.Q.D.N: (Fully Qualified Domain Name)

-A host name which ends with domain name is called FQDN.

To check FQDN

④ #Check -hostname ↴

It will read /etc/hosts.

Demo for DNS: # Pkginfo | grep -i sunbind

Server: Wstsun 80

# Cat /etc/services | grep -i domain

Create a domain name.

# domainname <domain name>  
sun.com

# domainname.

To save domain name permanent

domainname > /etc/defaultdomain.

+ Cat /etc/defaultdomain

# Svcs \*dns\*

# Svcs \*domain\*

# Svcadm restart domain

# mkdir /var/named

# cd /var/named

#. Prod

# vi named.hosts (forward lookup zone file).

Sun.Com. IN SOA wstsun80.Sun.Com. root.wstsun80.sun.com. (

2008300701

10800

3600

604800

86400 ).

Serial no.

Refresh time (in seconds)

(all these entries are for Secondary DNS)

Retry time (in seconds)

Expire (in seconds)

TTL (how many sec. cache memory has to save info).

Sun.Com. IN NS wstsun80.Sun.Com.

wstsun80 IN A 200.200.0.80

wstsun14 IN A 200.200.0.11;

www

# Prod

# vi named.rev (reverse lookupzone file)

0.200.200.in-addr.arpa. IN SOA wostsun80.sun.com. root.wostsun80.sun.  
com. (

2008300701

10800

3600

604800

86400 )

0.200.200.in-addr.arpa. IN NS wostsun80.sun.com.

80. IN PTR wostsun80

14 IN PTR wostsun14

:wq!

# vi /etc/named.conf

Options {

directory "/var/named";  
};

Zone "Sun.com"

{

type master;  
file "named.hosts";  
};

Zone "0.200.200.in-addr.arpa"

{

type master;  
file "named.rev";  
};

:wq!

#named-checkconf <

```
# Svcadm enable -y dns/server
```

```
# Svcs |* dns|*
```

## Client Side:

Client: wstsun14

```
#domainname
```

```
#domainname Sun.Com
```

```
#domainname > /etc/defaultdomain
```

```
# Svcadm restart domain
```

```
# vi /etc/resolv.conf
```

NameServer 200.200.0.80

domain Sun.Com

:wq!

```
# cp /etc/nsswitch.dns /etc/nsswitch.conf
```

```
# vi /etc/nsswitch.conf
```

goto 18<sup>th</sup> line : <sup>inter</sup>change

hosts: dns files

:wq!

```
# Svcs |* dns|*
```

```
# Svcadm enable -y dns/client
```

```
# Svcs |* dns|*
```

# nslookup ↵ (to check dns server)  
Services.  
→ wstsun14

### Server side

`# dig :`

(Used to diagnostics the errors)

Zone db files

`# dig Wtsun80.Sun.Com`

`# route add default <rooter ip> up` (to add gateway in sol.).

`# netstat -nr`

(display gateway ip address)

`# cat >/etc/defaultroute` (to <sup>Save</sup> ~~make~~ gateway Permanent).

20/10/09

### Trouble shooting:

- Check Main Config. file

`# Pkill -Hup named`

If Server Service is in Maintenance Mode

- ~~Svcadm~~ clear dns|server

`# named`

`# Pgrep named`

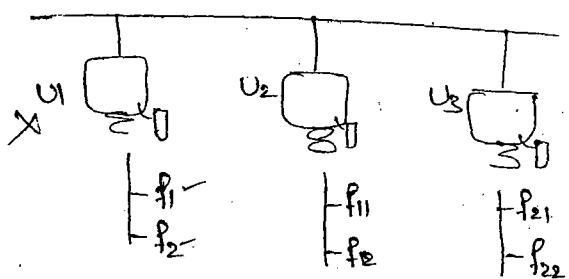
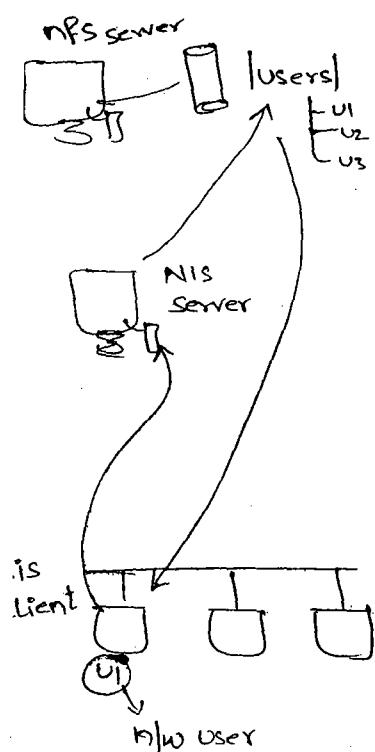
# NIS: (Network Information Service).

Present running

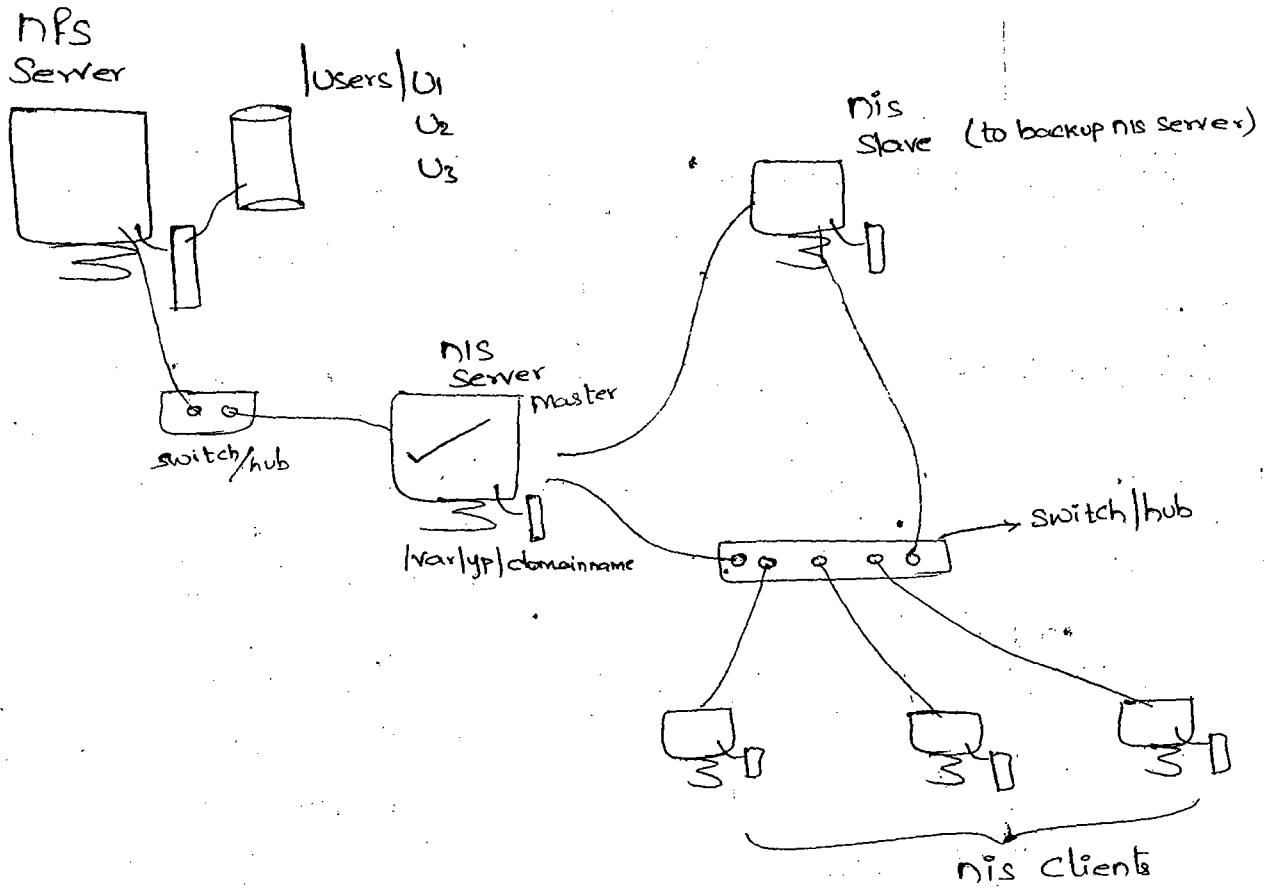
- NIS is used for mainly centralizes the users and users home directory and nw info. also.
- NIS runs based upon the autofs.  
↓  
(ondemand mounting)
- NIS follows flat domain concept. It won't support sublevel or multiple domains.

Windows — A.D.S (Active Directory Service)

Solaris — NIS (Network Info. Service).



- NIS maintains all the informations in terms of MAP's under **[Var]yp[domain name]**



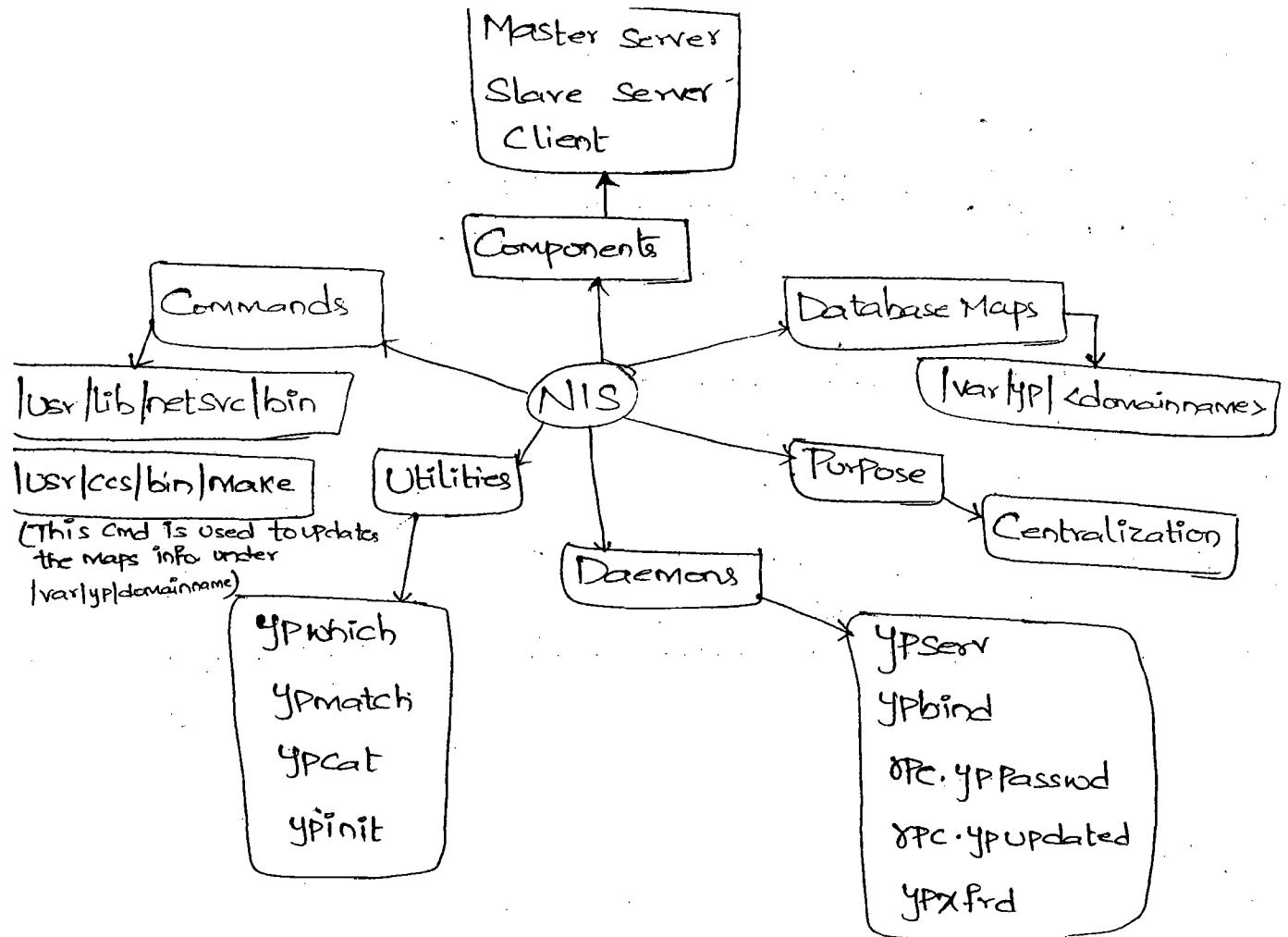
## Types of Components (NIS Components):

- NIS Master Server
- NIS Slave Server
- NIS Client Server.

## Packages for NIS:

: All the NIS Related Packages are located under 4/H cd

- SUNWYPR → (updates /root info)
- SUNWYPU → (updates /usr info)
- SUNWSPROT → (builds nis server)  
it contains bundles of Pkgs.



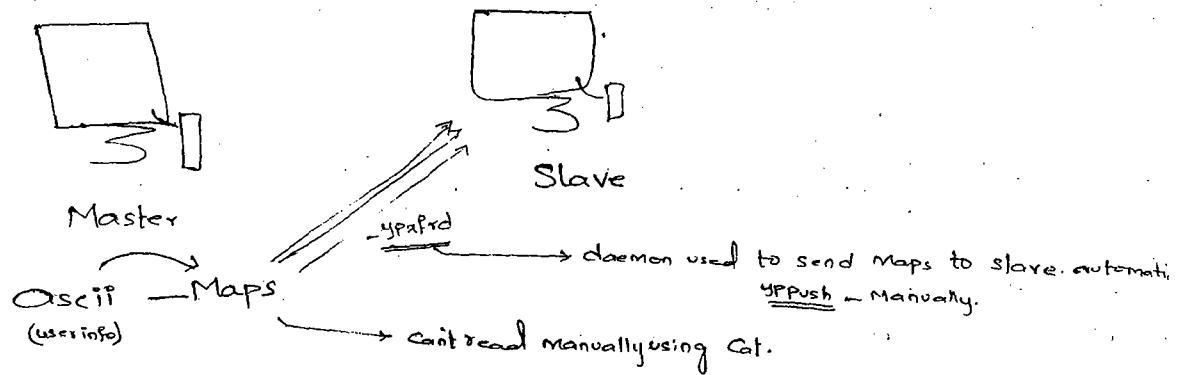
## NIS Master Server:

- Responsible for generating Maps and used to convert Ascii files to Maps.
- It Provides a Single Point of Control for the Entire NIS Domain.
- NIS Server is responsible for Provides the Authentication for NIS clients.

## NIS Slave Server:

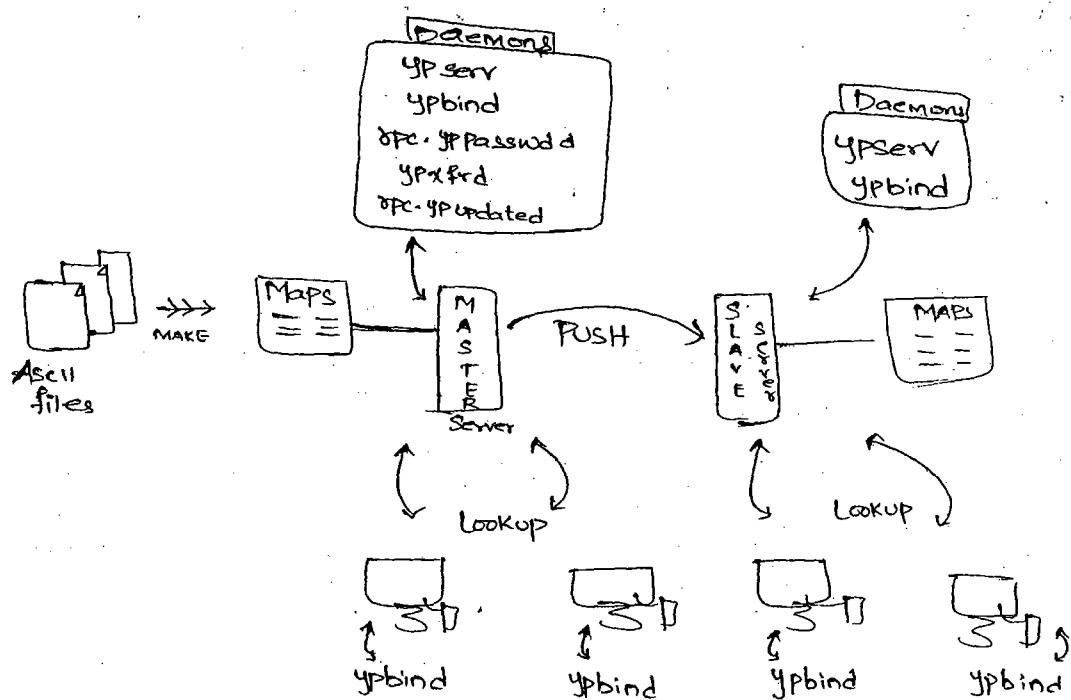
- Doesn't Contain any Original Maps but it Copies the Maps from NIS Master Server.
- It Provides availability (or) Redundancy When the Master Server crash.

- It provides backup depository for NIS Map info.
- Even it Provides load sharing.



NIS Client: - It doesn't contain any Ascii (x) Maps info.

- It dynamically rebinding to another server in case of Master server failure.
- **"ypbind"** is a daemon located in NIS client side which is responsible for sending the request to Master server and accessing the services.



## Daemons:

- Ypserv :- This daemon resides on NIS Master Server side as well as Slave Server.

- It answers the ypbnd requests from client side.
- It responds to client info. request.

Ypbnd :- It runs on all client systems.

- It makes initial Client - Server bind requests.
- It rebinds to another server, if the connection is lost with Master Server.
- It is responsible for provide communication b/w Master Server / Slave Server & Client.

Ypxfrd :- It runs only on Master Server.

- It generates in the Slave Server by using the **#Ypxfr** command to pull the maps (or) to push the maps from Master Server.

- It transfers maps at high speed.

- Ypc, Yppasswd :- It allows the users to change their passwords.

- It updates Password & Shadow files on the Master Server.
- Even it updates NIS Info. also.

- Ypc·ypupdatedd :- It is responsible for update the maps on the Master Server.

- Each and Every Maps Can have two files

• Pag      } (user's data)  
              } extensions.  
• dir      } (index for the maps)

Eg. Passwd.Pag , Passwd.dir , ethers.Pag , ethers.dir.

• Pag files Maintains Data. (all user info.)

• dir files Maintains Index. (Here we get the index for all users).

### Commands :-

# Ypcat : This Command is Responsible for reading NIS M

# ypwhich : Used to list NIS Master Server Name.

# ypmatch : - Searching the attr. (or) expressions which are located under Maps.

- Similar to 'grep' Command.

# ypinit : - This Command is used to initialize Master server | Slave server | Client.

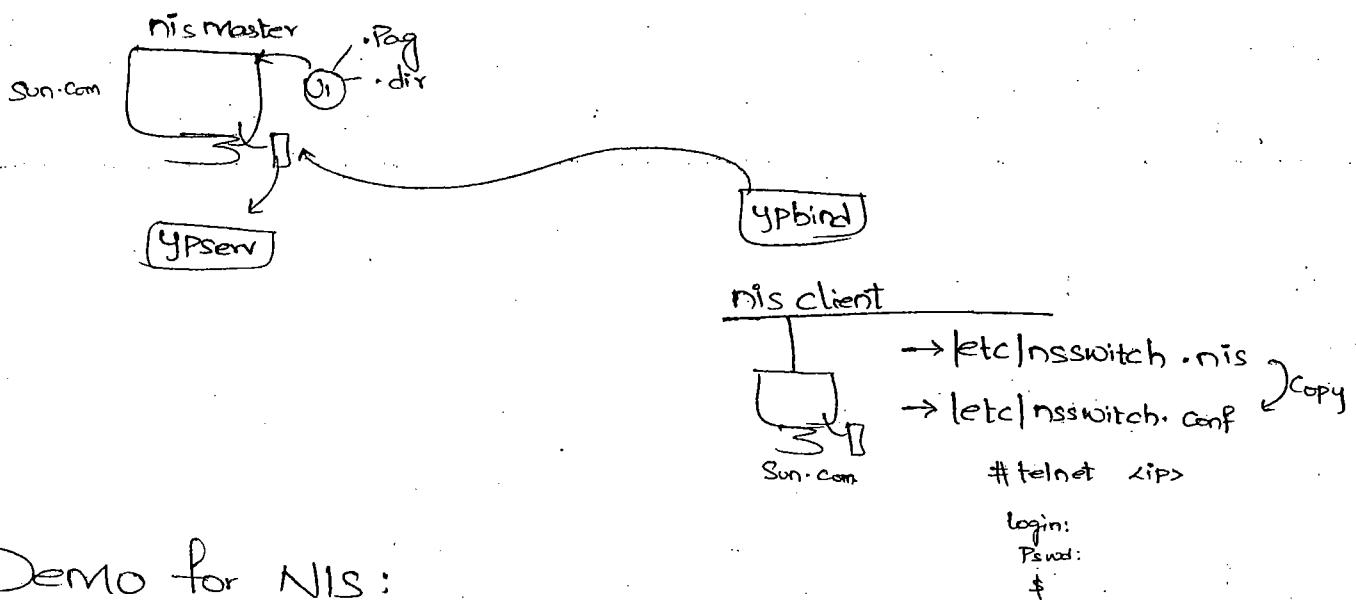
# ypinit -M : Used to initialise NIS Master server

# ypinit -S : " Slave Server

# ypinit -C : " client

- #yppush : - Responsible for Pushing Maps from NIS Master Server to Slave Server.
- Internally this cmd is responsible to run "ypaxfrd" daemon.

(NFS)



## Demo for NIS:

Master Server: hostsun80.

- Set the domainname.

```
# domainname Sun.com
```

```
#domain name >etc/default/domain (Permanent)
```

```
# Svcadm restart domain
```

```
# Svcs #domain/* (to check Service Configured (or) not)
```

Create a file & Mount it & Share it.

```
# chmod 777 /users
```

# Create some user accounts.

```
# useradd -d /users/sun1 -m sun1
# useradd -d /users/sun2 -m sun2
```

Assign Passwds

# Passwd Sun1

# Passwd Sun2

Open: vi /etc/dfs/dfstab ↵  
lastline

Share -f nfs /Users (or) Share -f nfs -o log /Users

↓  
if we want to generate  
any log msgs.

# Share all

# Share

## Implementing Indirect Autofs.

# Vi /etc/auto-Master

Lastline

/Users auto-home

:Wq ↵

# Vi /etc/auto-home

Lastline

\* hostsun80:/Users/ &

:Wq ↵

# automount -v

# svcadm restart autofs

# svcs autofs

Insert 4/4 Cd (to make server)

cd /cdrom/cdromo/Solaris-10/Product

#vi /etc/auto-Master

Make /home dir. Comment

# /home

:Wq ↵

Then Restart Autofs Service

```
# ls | grep SunWyp8  
# ls | grep sunwyp8  
  
# yes | Pkgadd -d .sunwyp8 sunwyp8 sunnspool
```

Precisely

```
# Pkginfo | grep SUNWyp8  
SUNWyp8  
SUNNspool (maintain bundled tools).
```

Making NIS Master Server: #vi /etc/hosts (Mandatory to mention Client IP)

200.200.0.199 wotsun199.  
twrd.

```
# cd /etc
```

```
# touch ethers netgroup netmasks timezone bootparams.
```

why?

Ethers :- Responsible for maintain ethernet id for clients.

Netgroup & netmasks : Responsible for maintain n/w info.

timezone : Responsible for maintain geographic status

bootparam : Responsible for maintain updates the bootable parameters when NIS client is logged in.

```
# yplinit -m
```

hostname  
slave host name.

```
# Is this Correct: Y.
```

```
# Quit on non-fatal errors: Y
```

All maps are generated under /var/yp/<domainName>  
sun.com.

for Confirmation

#ypwhich

#cd /usr/lib/netsvc/yp (maintains all NIS related cmds).

# ./ypstop

# ./ypstart (used to start deamon)

# Svcs -a | grep nis

-NIS Config. Completed.

### Client Side Config:

is Client: wotsun199

# Vi /etc/hosts (Mandatory to Mention Server IP)

200.200.0.80 Wotsun80

:wq!

# domainname Sun.Com

# domainname >/etc/default/domain.

# Svcadm restart domain

# Svcs -a | grep domain

# Cp /etc/nsswitch.nis /etc/nsswitch.conf ↴

# Vi /etc/nsswitch.conf ↴

34<sup>th</sup> Line

Automount: nis files

log

:wq!

Making hostsun199 to NIS Client

```
# ypinit -c  
host : hostsun199 :  
:
```

Is this Correct : y

```
# cd /usr/lib/netsvc/yp  
# ./ypstop  
# ./ypstart  
# yprhich  
# svcs -a | grep nis/client  
# listusers  
# cat /etc/passwd | grep sun1  
# cat /etc/shadow | grep sun1
```

Let us try to login as normal user (n10 user)

```
# telnet 0
```

troubleshooting

Login: sun1

Password:

If we don't get home dir. then troubleshoot

```
svcs -a | grep nis/client  
svcs restart nis/client.
```

If we add any new users we need map manually.

cd /var/yp

# /usr/ces/bin/make

Srcadm restart nis|<sup>Server</sup>Master

Check Packages  
if error.

# cd /usr/lib/netsvc/yp

21/10/09

## UnConfiguring the NIS :-

Server side :

① # Svcadm disable nis|Server ↴

# Cd /usr/lib/netsvc/yp ↴

② # ./ypstop ↴

(all)

# rm /etc/defaultdomain

# domainname " " ↴

# Svcadm restart domain ↴

③ # cd /var/yp ↴

# rm -r \*.\*time ↴

# rm -r Sun.Com ↴  
(domainname)

# cd binding ↴

# rm -r \* ↴

# cd .. | ↴

# vi aliases ↴

Sun.Com Sun.Com  
↓ Remove it.

vi /etc/auto\_master ↴

remove the Share Point entry's

vi /etc/auto-home ↴

remove the Share Point entries.

vi /etc/dfs/dfstab ↴

remove the Share Point entries.

#unshareall ↴

→ If you want to remove packages, here we can remove.

vi # Sys-unconfig ↴

[y] ↴

OK > boot ↴

Manually go through the options.

UnConfigure NIS Client:

# svcadm disable nis/client ↴

# cd /usr/lib/net/svc/yp ↴

# ./ypstop ↴

# rm /etc/default/domain

# domainname " " ↴

# svcadm restart domain ↴

vi /etc/nsswitch.files /etc/nsswitch.conf ↴

# Sys-Unconfig ↴

OK > boot ↴

## UnConfigure DNS:

Server Side:

```
# Svcadm disable dns|server  
dns|client  
# rm /etc/defaultdomain  
# domainname domainname ""  
# Svcadm restart domain  
# rm -r /var/named  
# rm -r /etc/named.conf
```

Client Side:

```
# Svcadm disable  
# cp /etc/nsswitch.files /etc/nsswitch.conf  
# rm -r /etc/resolv.conf  
# domainname " "
```

## Back Up :-

Why we need to take Backup?

- Accidental deletion of files, Powerfluctuations, H/w failures, logical errors i.e., Sector level Corruption, Any disasters.
- For Upgradations. Eg.: Patch Initialization.

<sup>Type of</sup>  
What files we need to take backup?

- All Critical/Manual filesystem.
- Main Config. files
- Database files.
- Some Command's O/P also.

Eg: #df -h  
# /etc/vfstab  
# /etc/system.

1. Traditional (Backup server)

2. Centralized (Having backup device on the Server. (a) Internal  
(b) external)

3. LAN free backup

4. Server free backup

(3 & 4 are SAN-based)

Classification

Backup Targets

1. Tape drive
2. Tape library
3. Optical CD-ROM
4. Local disk (Fastest backup)
5. NFS volume (Fastest & safe)
6. External disks

Backups

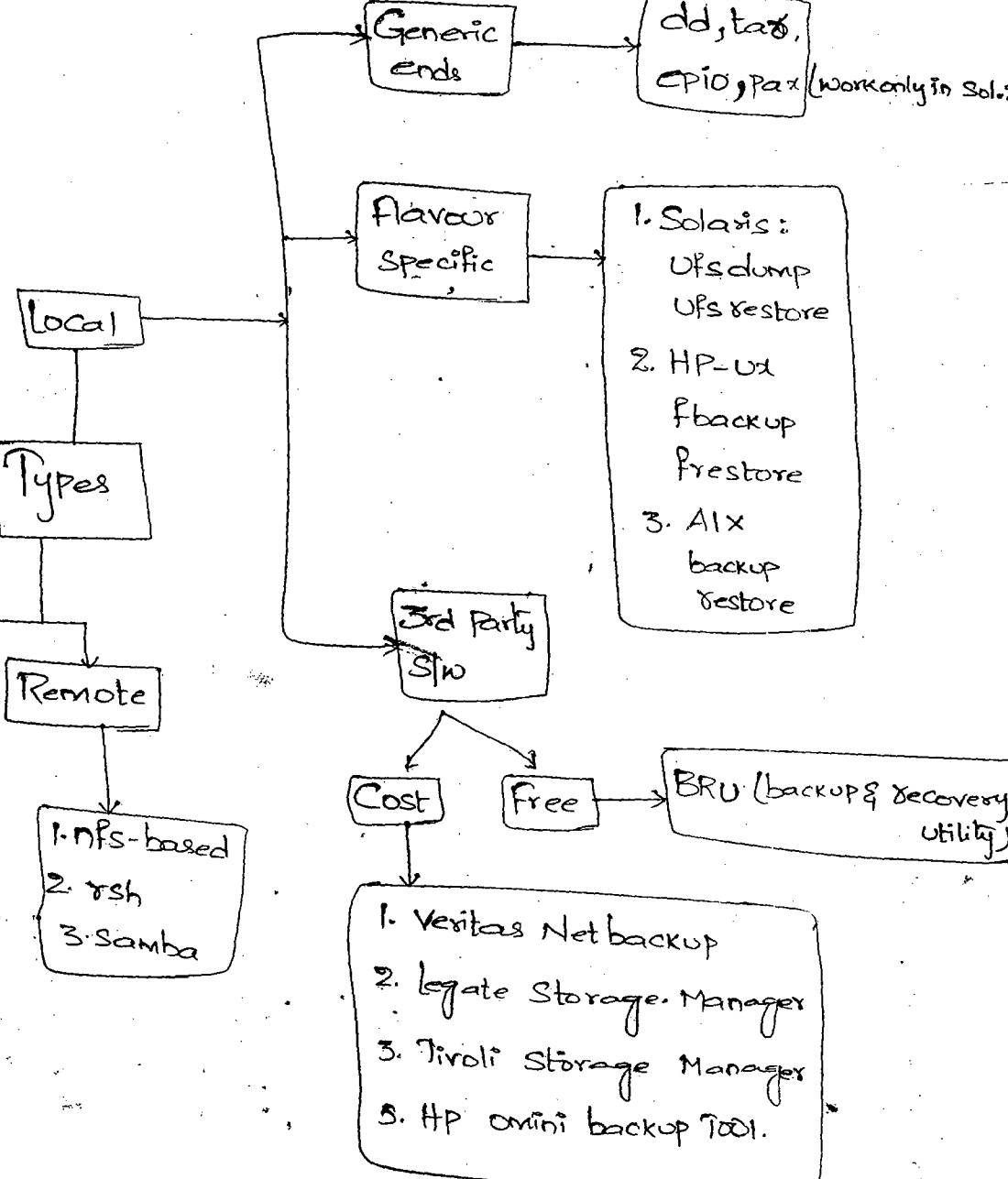
Types

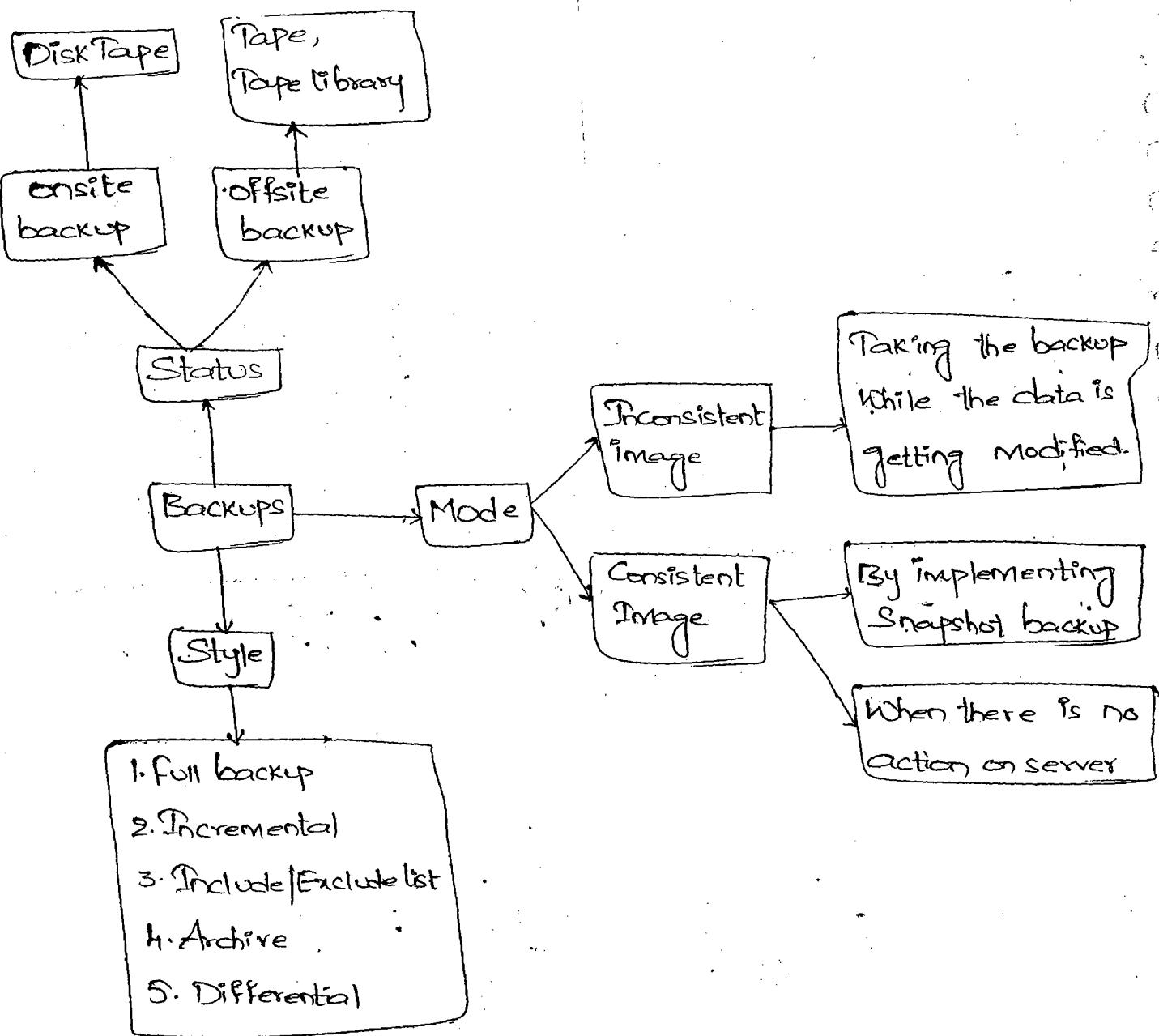
Network

Remote

Backup to tape  
Connected to LAN  
With an IP address

1. NFS-based
2. rsh
3. Samba





## Generic Backup's :-

- Can Support Any one of Unix Flavours.
- We Can't Perform Snapshot with the help of Generic Cmd's.
- Unix Doesn't Support any type of ~~extension~~

### Types:-

tar

Cpio

dd

Pack

### \*tar :- (tape archive)

- This Command will Work <sup>on</sup> any one of the Unix Flavours.
- We Can't Perform multiple Volumes backup.
- We Cannot take backup of device files.
- tar Syntax is easy to understand and easy to implement.
- Even we can perform update backup, include, exclude list backup.
- At Max. we can take 4gb to 8gb of data back.

syntax

```
# tar <options> <targetfile> <source>
```

<options>

- c → Create archive file
- f → Specify the target file
- v → Verbose
- t → list table of contents.
- x → Extract the backup
- X → Specify the exclude list file.
- I → Specify the include list file.
- u → update of backup.

J: # mkdir lwoil ↴

# cd lwoil ↴

# touch d1 d2 d3 d4 ↴

# tar -cvf ltar.bkp lwoil ↴

unix doesn't support  
any extensions. So  
manually we give .bkp,  
but to know what type  
of file it is, ↴

#file <filename> : displays what type of file it is.

#tar -tvf ltar.bkp

→ to see the backed up files.

# rm -r lwoil ↴

# tar -xvf ltar.bkp ↴

→ restore the dir.

## Update Backup:

In /tar.bkp day-1

/tar

/wil

/wil/d1  
  └── f1  
  └── f2  
  └── f3

/wil/d2

/wil/d3

day-2

I have created one more file under /wil/d1. now I want to update it.

(or) change data

# cat >> d1

abcd

Ctrl+d ↵

# tar -uvf /tar.bkp /wil ↵

↓  
existing file.

# tar -tvf /tar.bkp ↵

→

/tar  
/tar/wil  
/tar/wil/d1  
/tar/wil/d2  
/tar/wil/d3

Include list, Exclude list:

## CPIO :- (Copy input output)

- It is a dependent Command.
- We can perform device files backup also.
- It can support multiple volumes backup.
- When we take backup with the help of "CPIO" we can restore into any another location.
- We can take huge file system backup also.

Syntax:

# CPIO <options> <filename>  
archive

# Cmd | cpio <options> <filename>

always has to use  
with any cmd. only...  
not a single one.

- i → Input mode
- v → Verbose
- f → Specify the archive file
- t → list table of contents
- I → Specify the input file
- o → Specify output mode.
- O → Specify output file.

g: 1051/1234

# cd 1051 ↴

# ls

1 2 3 4

```
# ls | cpio -ovf -O |cp.bkp <
```

to know the backup performed (or) not

```
# cpio -ivt -T |cp.bkp <
```

- list table of contents.

to extract the contents.

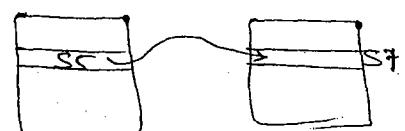
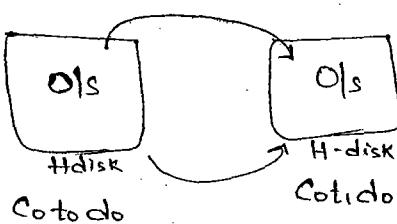
```
# cpio -iv -T |cp.bkp <
```

- Always use ls cmd to take backup
- If we use find cmd it will take from root dir. & we cannot restore in another place

## \* dd : (disk dump)

====

- generally we use this cmd for taking backup of filesystem → filesystem (or) disk → disk.



Syntax:

```
# dd if = <input file> of = <output file> bs = <size>
      ↓           ↓           ↓
      inputfile   outputfile.  blocksize
```

# dd if = /dev/rdsk/c0t0d0s1 of = /dev/rdsk/c0t1d0s1 bs = 2048 <

to check

```
# mkdir test & mount /dev/rdsk/c0t0d0s1 test &
```

```
#iostat -Cndx ↴
```

— Used to display disk level I/O operations.

```
#iostat -Cndx 1 10 ↴
```

— display 10 times the cmd O/P in 1 minute.

### Demo:

tar:

```
#mkdir /w1 ↴
```

```
#cd /w1 ↴
```

```
#touch 1 2 3 4 5 ↴
```

```
#ls ↴
```

```
#cd ↴
```

```
# tar -cvf /tar.bkp /w1 ↴
```

```
# tar -tvf /tar.bkp
```

```
# rm -r /w1 ↴
```

```
# tar -xvf /tar.bkp
```

```
# cd /w1 ↴
```

```
# ls
```

— We Cannot restore the dir. info to another location using tar cmd.

— We Can restore files under that dir. but not dir.

```
# tar -cvf /tar.bkp2 . ↴
```

```
# cd /testqq
```

```
# tar -xvf /tar.bkp2 ↴
```

```
# tar -cvf |tar.bkp |wii
```

```
# tar -tvf |tar.bkp
```

```
# cd |wii|
```

```
# rm *
```

```
# ls
```

### Performing Include list extraction

```
# cat > |include|
```

```
|wii|/1
```

```
|wii|/A
```

```
|wii|/3
```

```
Ctrl+d ↴
```

```
# cat |include| ↴
```

```
# cd |wii| ↴
```

```
# ls
```

```
# tar -xvf |tar.bkp -T |include| ↴
```

```
# ls ↴
```

### Exclude list extraction:

```
# cd |wii| ↴
```

```
# rm * ↴
```

```
# ls
```

```
# cat > |exclude| ↴
```

```
|wii|/3
```

```
|wii|/5
```

```
Ctrl+d ↴
```

```
# tar -xvf |exclude| |tar.bkp| ↴
```

```
# ls.
```

## Performing backup:

How to Perform Include backup?

```
# cd /var
```

```
# ls
```

```
# cat | include
```

```
# tar -cvf /tar.bkpq -T | include
```

```
# tar -tvf /tar.bkpq
```

## CPIO:

```
# cd /var
```

```
# ls | cpio -ovf -o /cp.bkp
```

```
# cpio -ivf -T /cp.bkp (to see)
```

```
# cd /var
```

```
# rm *
```

```
# cpio -ivf -T /cp.bkp
```

```
# ls
```

```
# find /var
```

```
# find /var | cpio ...
```

try to take backup & using find &  
restore in other location.

## dd:-

```
# Create a f/s & Mount it.
```

```
# Create another slice. don't build f/s & don't mount.
```

```
# dd if=/dev/rdsk/c0d0s5 of=/dev/rdsk/c0d0s6 bs=2048
```

## \* Pack :-

- Pack is a command used to Compress only files data.

- not for directories.

- all Pack files extension is

" .Z "

```
# Vi Sunn <
```

:-

:wq <

```
# du -sh | Sunn
```

1K | Sunn

```
# Pack -f | Sunn
```

```
# du -sh | Sunn
```

```
# ls
```

```
# file Sunn.z
```

## Unpacking: (Uncompressing packed file)

```
# Unpack Sunn.z
```

```
# ls
```

```
#
```

zip

## Lab Project:

### Anonymous FTP:

- without Authentication we can login into the Server & we can transfer files.

here we can get files only from Pub directory.

Server:- (200.200.0.1)

= ftpConfig | anonymous ↵ : used to create configuration directory of ftpserver.

= cd | anonymous ↵

: This directory contains bin,  
dev,  
etc,  
lib,

Platform  
Pub  
usr

↳ ls ↵

= cd Pub

↳ ls ↵

↳ touch a b c d ↵

Client:-

telnet 200.200.0.1

# ftp 200.200.0.1

Name: anonymous

Password: ↵

↳ ls ↵

ftp> cd Pub

ftp> ls

ftp> mget a b c ↵

ftp> bye ↵

### Server:

```
# cd anonymous  
# mkdir incoming  
# chmod 777 incoming
```

### Client:

```
# ftp 200.200.0.1
```

Name: anonymous

Passwd:

```
ftp> ls
```

```
ftp> cd incoming
```

```
ftp> put apple
```

```
ftp
```

→ # vi /etc/ftpd/ftpd.access  
↓  
goto 49th line

:49

~~remove~~ comment that line.

21/10/10

## Types of Backup's:-

1. Local Backup

2. Remote

3. Network

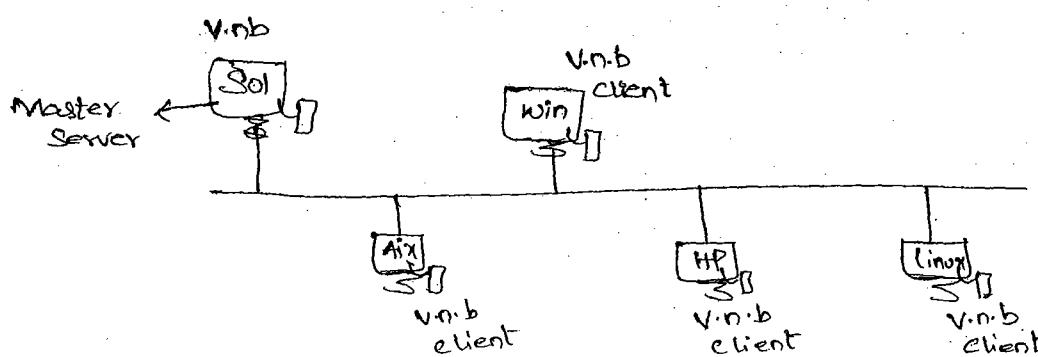
1. Local Backup: Taking the backup of local System <sup>internal</sup> or external disks.

2. Remote Backup: Taking the backup of local System being in Remote Machine.

3. Network Backup: We can take the backup of heterogeneous Operating Systems.

- N/w backup can support different types of Operating System backups.

Eg: Symantec → Veritas Net Backup (V.n.b)  
HP → Omni  
IBM → Tivoli



## Levels of Backup:-

Tape drives  
 ↓  
 Tape cat.  
 ↓  
 24 GB.

1. Incremental Backup
2. Cumulative Backup
3. Differential Backup
4. Full Backup

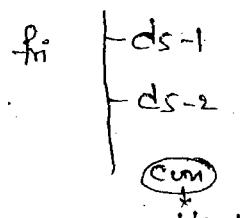
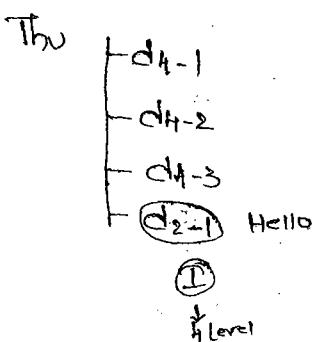
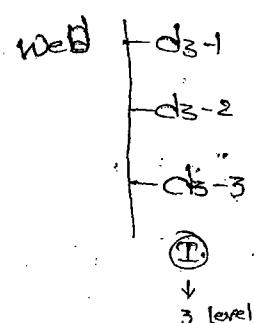
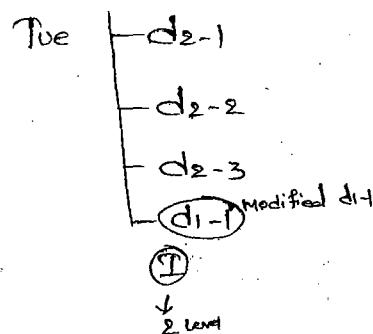
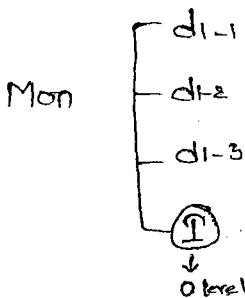
- By using flavour specific backup we can use all these levels.

## Flavour Specific Backup:-

# Ufsdump  
 # Ufsrestore

} only in Solaris.

✓ Daily Incremental  
 ✓ Weekly Cumulative



BKPUP → 0 1 2 3 4 5 6  
 ↓ ↓ ↓ ↓ ↓ ↓  
 full cum 7 8 9.

- In full backup only we get all Incremental backups.
- Cumulative backup has only some ~~all~~ incremental backs.

## Tape Technologies:

- DDS (Digital Data Storage)
- DAT (Digital Audio Tape)
- DLT (Digital linear Tape)
- LTO (Linear Tape open)

## Tape Management Commands:

#mt St ↵ : Used to list Magnetic tape status.

#mt fsf ↵ : to move one tape level forward.

# mt fsf 2 → two tape levels forward.

#mt nbsf ↵ : to move one tape level backward.

#mt nbsf 2

#mt rew ↵ : to Rewind the tape. It automatically placed in 0<sup>th</sup> level.

#mt erase ↵ : Used to remove the data on Tapelevels.

#mt off ↵ : to eject the tape (or) taking to offline.

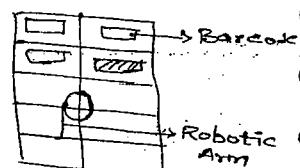
## Tape drive device Path:

/dev/rmt/0

```
st: devfsadm -c tapedrive 2  
-t detect.
```

Tape library:  
To take backups of very huge amount of data.

SL-8500



# Ufsdump:

## Syntax:

**# Ufsdump** <options> <target> <Source>

- c → Create a backup
  - u → Update "etc/cdumpdates" file
  - v → Verbose
  - f → Specify the target

# UFs restore :-

## Syntax:

`# ufs restore <options> <target>`

- x → Extracting
  - v → Verbose
  - f → Specify the target file
  - t → table of Contents
  - i → interactive Restore.

卷之二

```

graph LR
    A[ufsdump] --> B[ocufv]
    A --> C[ldevlrmto]
    B --> D[Level of backup]
    C --> E[destination]
    B --> F[users]
    C --> F

```

|etc |dumpdates:

dumpdates: — This is the Main Config. file for filesystem level backups.

- It Maintains info like level of backup, file System Name, Timestamp.

```
# UFS restore -xvf /dev/rmt10 <
```

## UFS Snapshot's :-

- Snapshots are mainly used for Performing Online backups.
- Snapshot we can take only on filesystems.
- only one Single Snapshot for one Single filesystem.
- Always Snapshot are read only.
- The default Snapshot device Path is

/dev/fssnap10  
|  
2  
:  
1

## Commands Related to Snapshot:

**#fssnap** : is a command used to Create a snapshot on file System

**#fs snap -i** : used to display snapshot info.

**#fs snap -d** : Used to delete a Snapshot

**#fssnap -f ufs -o bs=1var|sun |users**  
GP: /dev/fssnap10.      ↓      ↓  
                              backend store      Buffer

- We cannot read Snapshot info. directly. So, to read

```
#mkdir testsnap
```

```
#mount -f ufs -o ro /dev/fssnap10 testsnap <
```

## DEMO for -Flavour Specific Backup:

Create a file & Mount it.

```
# cd /users
```

```
# ls
```

```
# touch d1-1 d1-2 d1-3
```

```
# Cat > d1-1
```

1234

Ctrl+d ↴

```
#
```

Connect the tape drive physically to System & Insert D.D.s Cartridge.

```
# devfsadm -c tape ↴
```

```
# cd /dev/rmt
```

```
# ls
```

} to know tape drive detected (or) not

```
# mt st ↴
```

```
# ufsdump -O curf /dev/rmt10n /users ↴  
↓ nowwind
```

```
# cat /etc/dumpdates ↴
```

```
# mt st ↴
```

```
# cd /users
```

```
# touch d2-1 d2-2 d2-3
```

```
+ Cat >> d1-1
```

abcd

```
# Ufsdump -2uvf /dev/rmt/0n /users
```

```
# Cat /etc/dumpdates ↴
```

```
# Mt st ↴
```

```
# Cd /users ↴
```

```
# Ls
```

```
# touch d3-1 d3-2 d3-3 ↴
```

```
# Ufsdump -3uvf /dev/rmt/0n /users ↴
```

```
# Cat /etc/dumpdates
```

```
# Mt st ↴
```

```
to see all backup info.
```

```
# Ufsrestore -tvf /dev/rmt/0n  
error
```

```
# Mt nbsf
```

```
# Mt st
```

```
# Ufsrestore -tvf /dev/rmt/0n
```

```
# Mt mt fsf
```

```
# Mt st
```

```
# Cd /users
```

```
# touch d4-1 d4-2 d4-3
```

```
# Ufsdump -1uvf /dev/rmt/0n /users ↴
```

```
# Cat /etc/dumpdates
```

```
# Mt st
```

```
# mt nbsf
```

```
# Ufsrestore -tvf /dev/rmt/0n
```

```
# mt st
```

```
# Ufsrestore -tvf /dev/rmt/0n <
```

Performing full backup.

```
# Ufsdump -Ouvf /dev/rmt/0n /users <
```

```
# cat /etc/dumpdates <
```

```
# mt st
```

```
# mt nbsf
```

```
# Ufsrestore -tvf /dev/rmt/0n <
```

```
# cd /users < -- Restore the files.
```

```
# rm * <
```

```
# mt st <
```

```
# mt nbsf
```

```
# mt st
```

```
# Ufsrestore -xvf /dev/rmt/0n
```

Specify next Volume #: 1

Set orientation [yn]: y

```
# cd /users :y
```

```
# rwd
```

```
# ls
```

## How TO Perform Interactive Restore?

```
# cd /users ↴  
# rm d2-1 ↴  
# rm d2-2 ↴  
  
# mt st ↴  
# mt nbsf 4 ↴  
# mt st ↴  
# UFSrestore -tvf /dev/mt0n ↴  
# mt st ↴  
# Mt nbsf ↴ again to comeback  
# mt st ↴  
  
# UFSrestore -ivf /dev/mt0n ↴  
  
UFSrestore > ls ↴  
UFSrestore > add d2-1 d2-2  
UFSrestore > ls ↴  
UFSrestore > Extract ↴  
Extract  
if only d2-1 is required  
UFSrestore > Delete d2-2 ↴  
UFSrestore > extract ↴  
UFSrestore > quit ↴  
  
# cd /users  
# ls
```

## Snapshots Demo

Create files & Mount it

Create some files & dir on that mount point.

```
# cd /users ↵
```

```
# fssnap -F UFS -o bs=/var/wiig /users ↵
```

```
# mkdir /testsnap ↵
```

```
# Mount -f UFS -o ro /dev/fssnap10 /testsnap ↵
```

```
# df -h ↵
```

```
# cd /testsnap ↵
```

```
# ls ↵
```

```
# cd /users ↵
```

```
# ls ↵
```

```
# UFSdump -Ocuf /dev/rmt10n /testsnap ↵
```

23/10/09

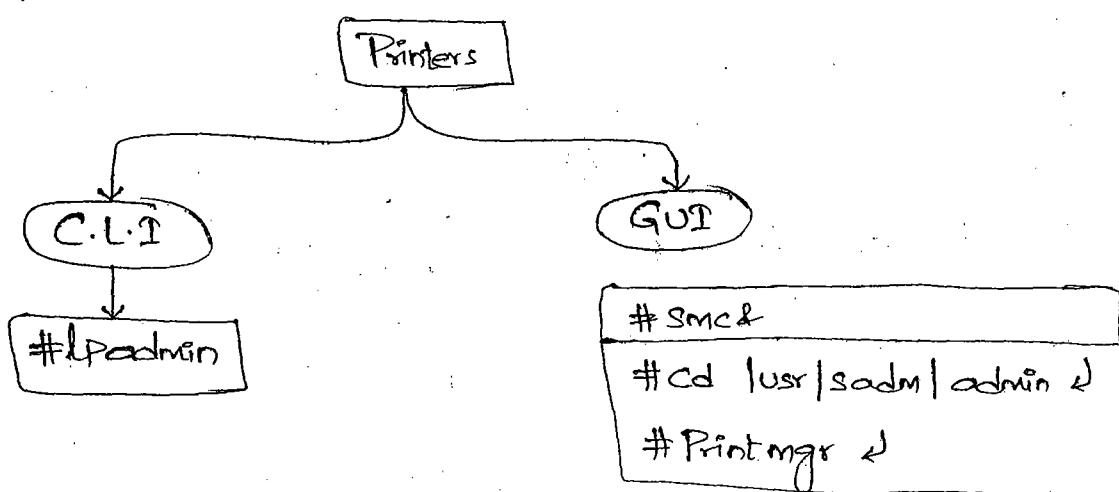
decisions  
Configure  
cmds.

## PRINTERS :-

- Printer's administration enables you to Manage the Printers, Configure, adding the Printer's to Services.
- Solaris operating Environment can Support different types of Printers.
- ```
#cd /usr/lib/lp/bin
```

 } : To know how many printers it can support.  

```
# ./lpmarks
```
- We can Configure the Printers in a two ways
  1. Command line Interface
  2. Graphical User Interface



To install the total Pkgs from cd.

```
#cd /cdrom/cdromo<br/>#ls<br/>R... Sol....<br/>... installer<br/># ./installer <br/>
```

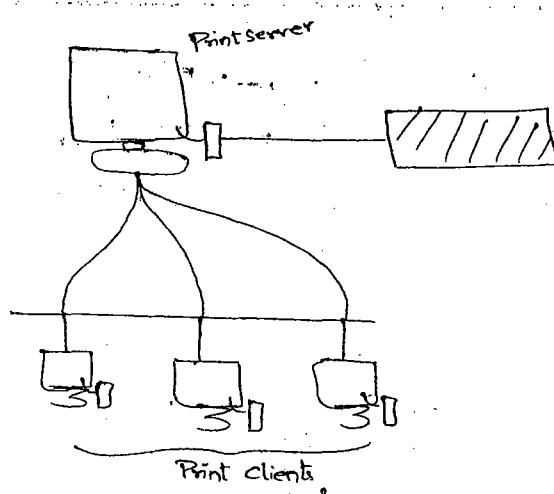
We have two types of Printers

1. Print Server
2. Print Client

1. Print Server: - Is Configured & Managed by the One Single System

- It is Responsible for Providing Spooling Requests to Client.

2. Print Client: - Is Responsible for Sending Print Request to Print Server.



- Solaris Operating Environment Can Support 3 types of Printers

- Local Printer
- Remote Printer
- N/W Printer.

1. Local Printer: These type of Printers are Configured by One Single System & accessing by that System.

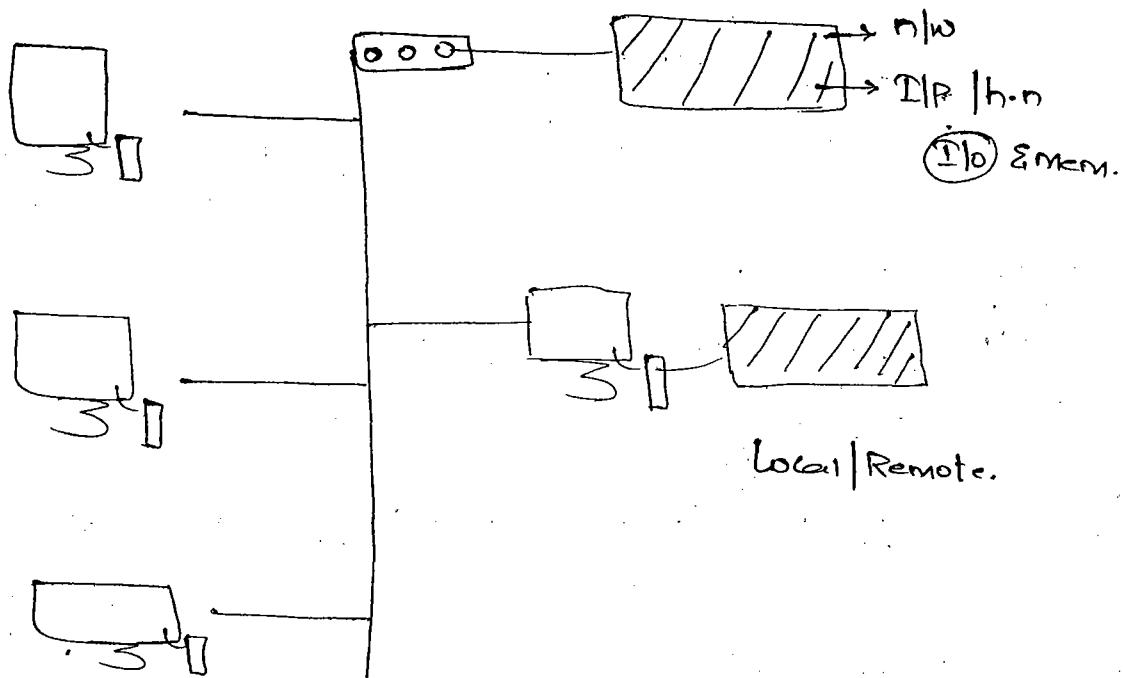
We have 2 diff types of Comm. in Printers

- Serial
- Parallel

2. Remote Printer:- The Printer devices are directly connected to Remote System in n/w. It has unique IP address & hostname etc.

3. Network Printer:- These types of Printer devices are directly connected to Switch/hub.

- The n/w Printers can support heterogeneous platforms.

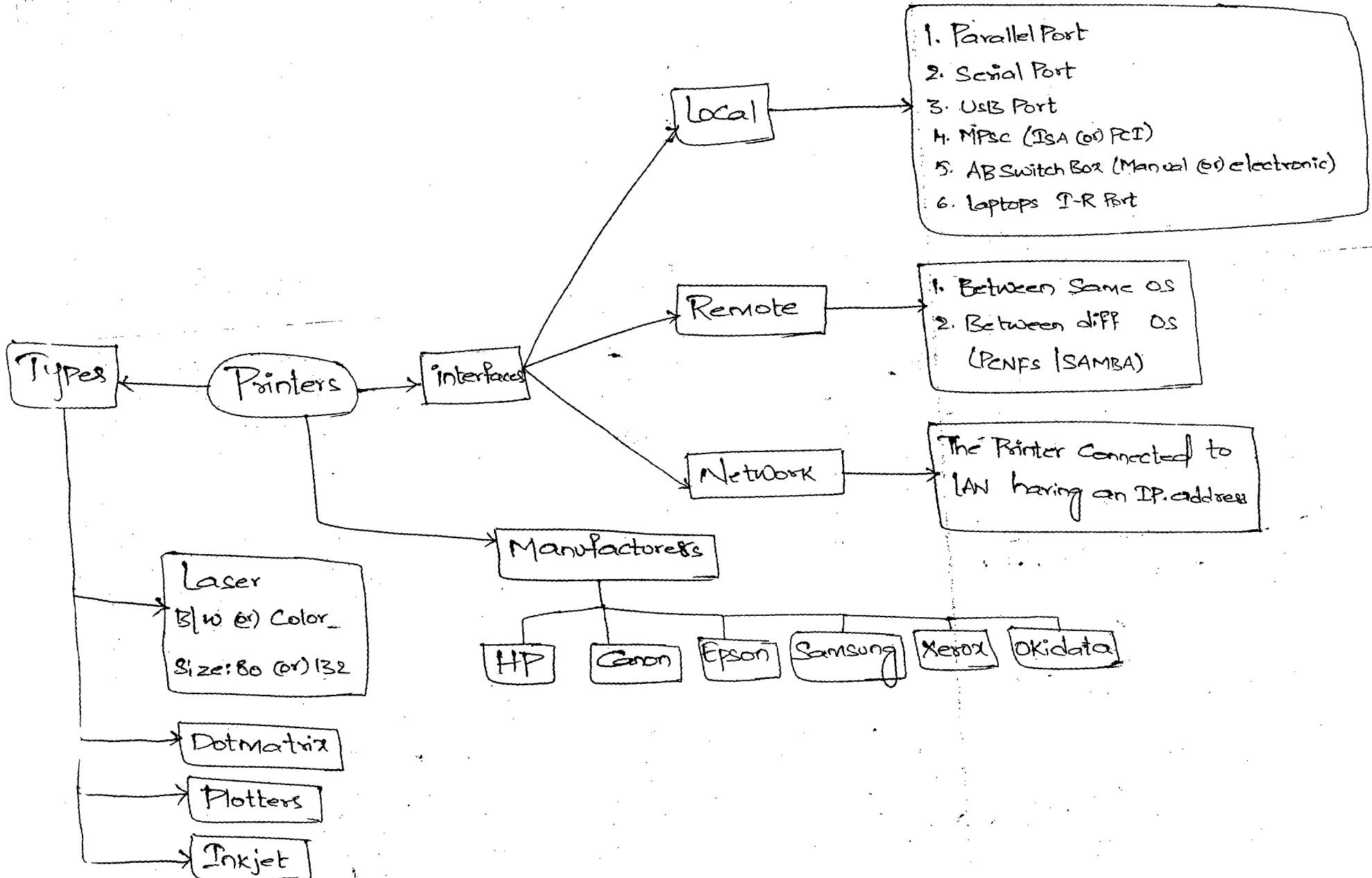


### Deamons for Printers:

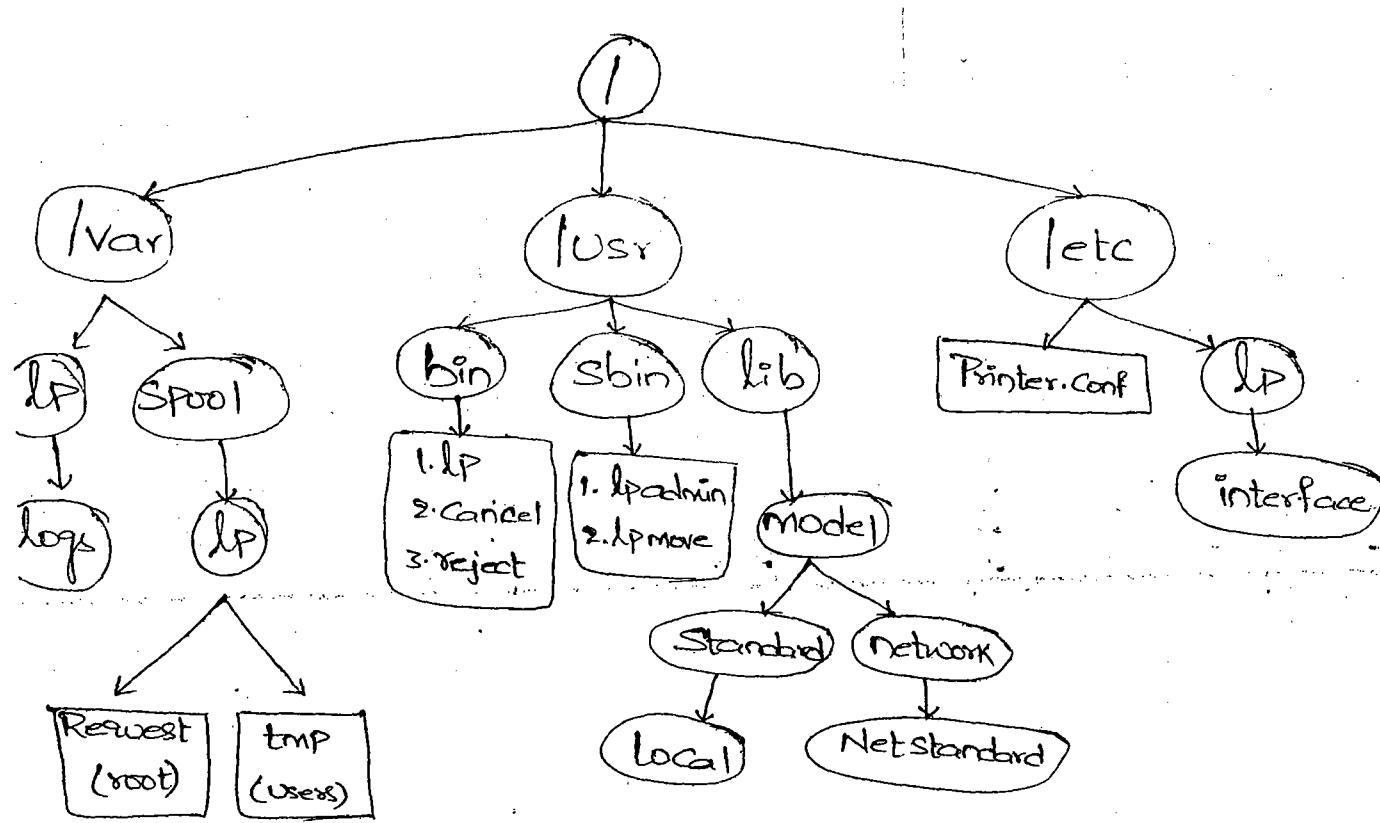
- in.lpd
- lpsched

svcs -a | grep Print

- Printer info.



## Commands & Log's location for Printers:



- Printers can use a different type of language **PPD**.

(Post Print descriptor Language). This language vendor is Adobe.

## Commands related to Printer:-

- All Printer Cmds. are located under

```
# cd /usr/lib/lp/local
```

- |           |                            |            |
|-----------|----------------------------|------------|
| → lpadmin | → enable                   | → lpmov    |
| → lpstat  | → accept                   | → lpshut   |
| → lpmov   | → <del>cancel</del> lpshed | → lpsystem |
| → reject  | → disable                  |            |
|           | → lp                       |            |

## Files Related to Printer :-

**/etc/printers.conf**

:- This is Main Configuration file for Printer.  
It maintains system wide local (or) Remote  
Printer info. with the logical names.

**/etc/lp/interfaces**

:- /<logical name of the printer>  
"SUNP"

- When we configure a Printer it automatically  
Creates a file under this location.

- By default the interfaces directory is empty.

**/usr/lib/lp/model/standard**

- When we config. local Printer the script  
file (standard) is copied to interfaces directory.

**/usr/lib/lp/model/netstandard**

- for n/w Printers.

**/etc/lp/printers/SUNP |**

↓  
logical Name

**# Cat User.deny**

**# Cat User.allow**

! By default all "Normal" users can  
also access the Printer devices. If you want to restrict  
any user, we have to provide User entry Under  
User.deny file.

- If we modify these files, we need to restart the Print  
Services.

- Manually we need to Create User.allow file.

→ For ex if the same user exists under User.deny & User.allow  
then by default it goes to User.allow file.

# cd /usr/lib/lp/bin : It Contains binary executable

files related to Printers.

- Eg: # ./lpstat : Prints all supported Printers.

# cd /usr/lib/lp/local : It Contains all binary executable

commands related to Printer.

# cd /var/lp/logs : It Maintain all Printers related logs.

# cd /var/spool/lp/requests : It Maintain the info.

of Print request which are in queue.

# lpq : i

Demons:

# lpsched :- It's a line printer schedule (or) daemon. Responsible for scheduling the print jobs.

- This daemon is for local printer.

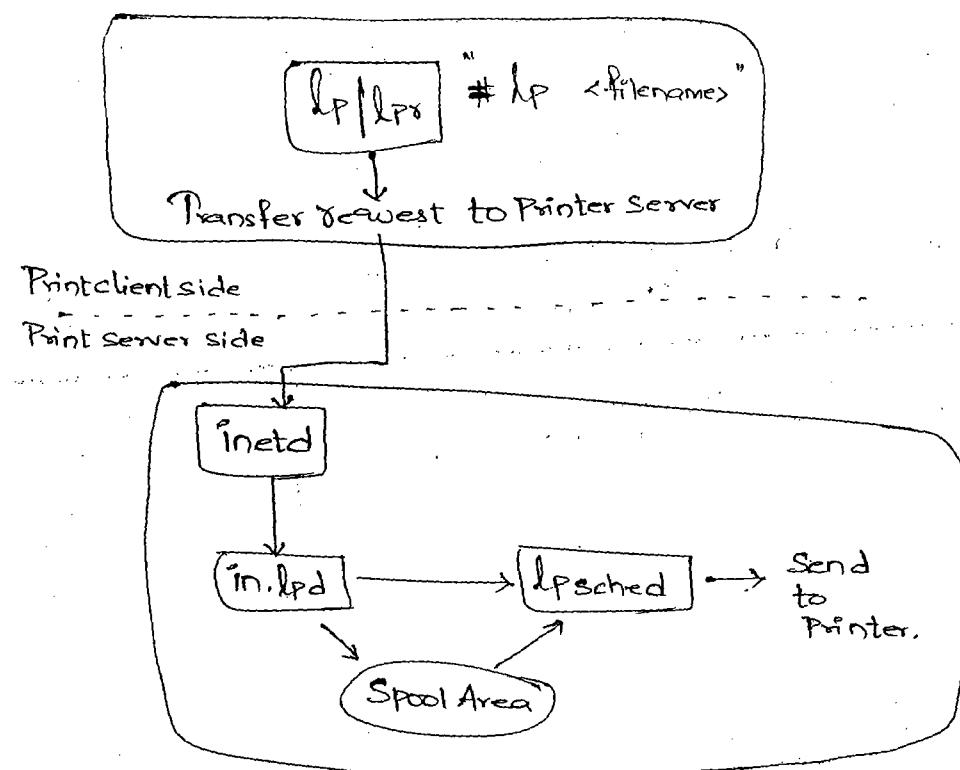
- It updates the info. in system files about printer setup configuration.

- It also manages the requests issued by the lp command.

- This daemon started by the control script

/etc/rc2.d/S80lp : Starting Script.

- in. lpd :- inetd daemon is responsible for starting the in.lpd daemon.
- It is responsible for providing the services to n/w Printer.



### Demo:-

- Connect the Printer device Physically.

```
# devfsadm -c Printers
```

Device Path for Printers

```
# cd /dev/Printers/0
```

To check Connection is Proper or not

```
# banner sun >/dev/Printers/0
```

Configuring the Printer in Command level.

```
# lpadm -P Sunif -v /dev/Printers/0
```

↓                    ↓  
logicalname      Specifiddevicepath

## Setting | enabling The default Printer.

```
# lpadmin -d SunP ↵
```

```
# lpstat -d ↵ : Used to display the default Printer.
```

Once you make default Printer check for logical name.

```
: cd /etc/lp/interfaces
```

```
# ls ↵
```

```
# cd /usr/lib/lplmodel/ ↵ : This info. is Copied to SunP
```

```
# ls Standard
```

```
: ls netstandard ↵ : for new printer.
```

"enable" → only works in Bourne shell

```
# exit
```

```
# enable SunP ↵
```

```
# accept SunP ↵
```

## To Print

```
# lp <filename> ↵ : it will also print banner info.
```

```
# lp -o nobanner <filename> ↵ : it prints only data.
```

```
# lp -o banner <filename> ↵ : Prints banner also.
```

To make permanent that banner should not display.

```
# cd /etc/lp/interfaces ↵
```

```
, vi SunP ↵
```

go to line no: 332 (change no to yes)

• nobanner = "yes"

```
:wq ↵
```

How to Restrict a normal User without access to Printer.

Create Normal User & Assign Password.

# telnet o ↴

(o)

# su - ↴ login as: normal user.

\$ lp file1 ↴ : Prints file1.

Restring the Normal User.

cd

# /etc/lp/printers/supp.

# cat > users.deny

(Put) <> username >

# svcs -a | grep Print

# Svcadm restart svc:/application/print/server:default.

# svcs -a | grep Print ↴

# su -

\$ cat file1 ↴

\$ lp file1 ↴

error: you can't Print

\$ lpstat -P ↴ to check Printer service to that user.

UnConfiguring The Printer:

# lpadmin -x supp ↴

: When you execute this it automatically removes entry from /etc/lp/interfaces, /etc/lp/printers.

# lpstat -s ↴ : Used to display line Printer statistics. 128

Basics →

Intermediate System Admin → SA - 200 - S10

Advanced System Admin → SA - 202 - S10

Trouble Shoot → Solaris Common Messages & Troubleshooting guide.

# lpstat -t : displays including time Modifications.

# cd /usr/lib/lp/bin  
+ ./getmakes : used to display supported Printer devices.

# cd /var/spool/lp/requests : responsible for maintaining ongoing requests.

How to take Printout with the help of ftp Service?

(If it is N/W Printer).

- Check Printer IP address is available (or) not.

ftp <ipaddress>

Login:

Password:

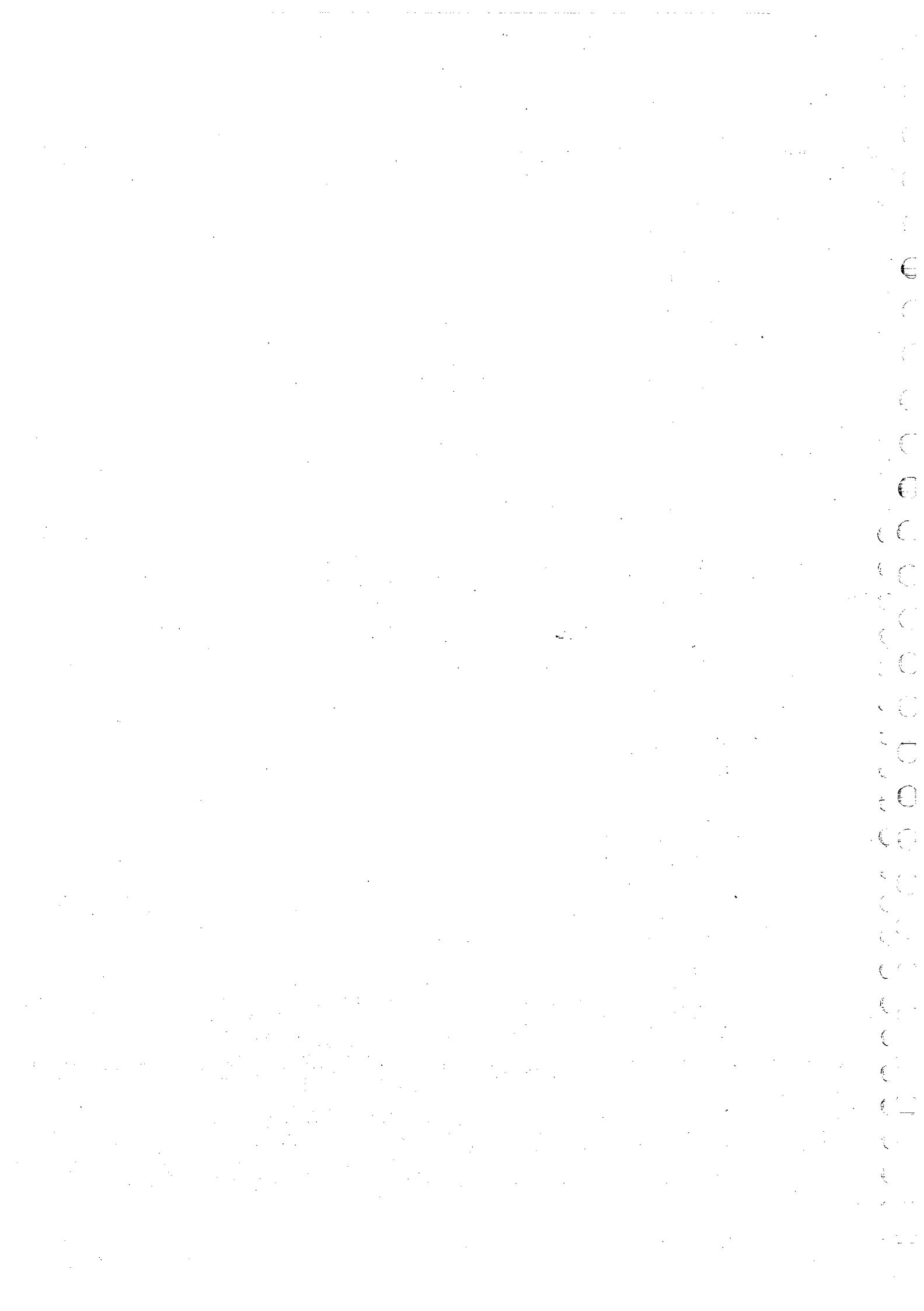
ftp> Put <filenames>

#lp -P <Printename> <filename> ↳ : TO print if multiple Printers available.

నుండి

Solaris 10  
- Bill Calkins

/usr/dt/bin/dtlogin



24/10/01

## Security Issues :-

- How to Restrict a User Even though he know the Root Password of my System.

vi /etc/default/login ↴

UnComment

\*# Console = log

Ctrl+d

- To Restrict all the normal users to login into My System.

Change Root Password / lock the Password.

(or)

# Passwd -l <username> ↴

for all users

# Passwd -l \* ↴

(or)

# cd /etc ↴ it checks.

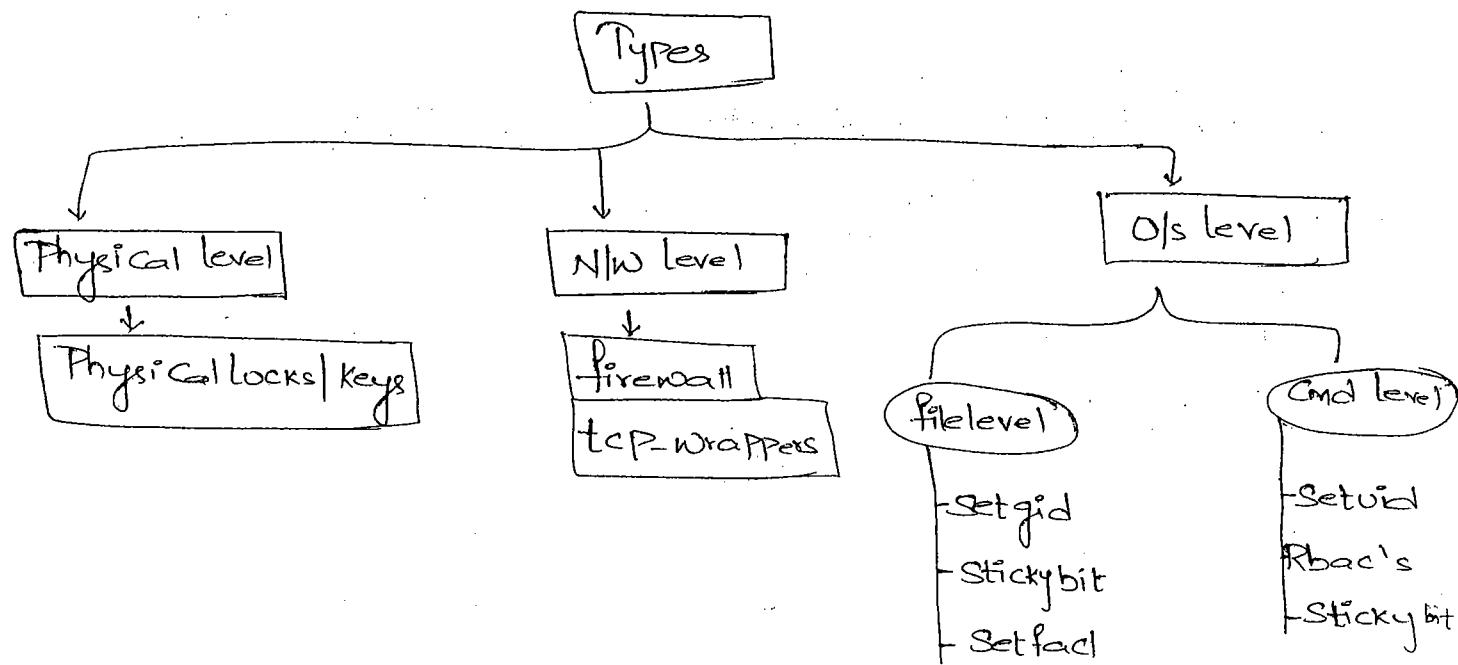
# touch nologin ↴ : no user can login. /etc/passwd, /etc/shadow

# vi nologin ↴

You cannot login at this Particular time.

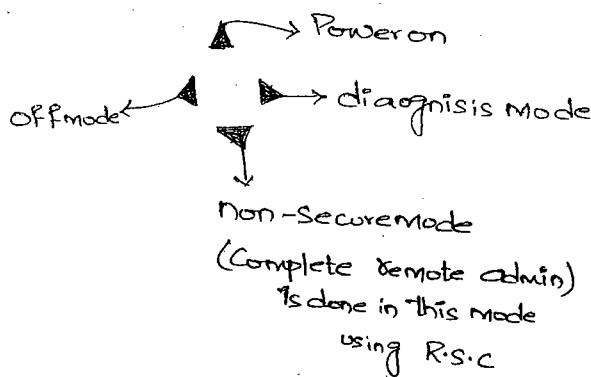
:wq ↴

## Types of Securities :-



## Physical level :-

In (E-250) Server.



- Using Rsc we can go through bios level of Remote Machine also.
- only If users can be authenticated using R.s.c.
- Entire Server resources will be available.
- The advanced to Rsc is clones. clones is used in Present Servers like (V250 & etc.)

## N/w level :-

tcp-wrappers : by default it is false.

to change anything in N/w level we should make  
tcp-wrappers equal to True.

#inetadm is a daemon for tcp-wrappers.

To restrict n/w level users we have two files.  
host.deny  
host.allow

## OS level :-

### 1. Command Level :-

Setuid :- By implementing Setuid, we can assign Super User Commands to a normal User.

To Know the Absolute Path of a Command

# which <Command>

### Implementing Setuid :

#### 1. Symbolic mode

# chmod u+s /usr/sbin/format

#### 2. Numerical mode

# chmod 4755 /usr/sbin/format

assigning the setuid to users.

```
# ls -l /usr/sbin/format
```

-r-s/r-x/r-x  
↓  
Setuid

Then goto User mode.

```
$ /usr/sbin/format
```

To disable Setuid

```
# chmod u-s /usr/sbin/format
```

```
# chmod 555 /usr/sbin/format
```

disadvantages:

- Using Setuid all normal users can access that command. We can't give only to a particular user.
- we can give to a particular user using Rollbacks (Rbac's).

Permissions for /etc/shadow is 400

/etc/passwd is 644

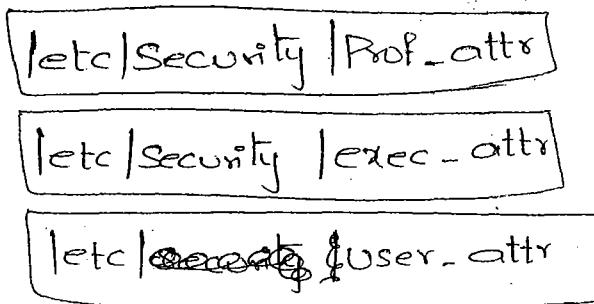
```
# ls -l /usr/bin/passwd
```

-r-s/r-s/r-x

by default /etc/passwd file have Setuid.

## Role based access Control list : (Rbac's)

Main Configuration files related to Rbac's



### /etc/security/prof\_attr :-

This file is responsible for maintaining profile name.

Profile name.

# vi /etc/security/prof\_attr ↵

foto last line

Profmgmt :::  
:wq ↵

### /etc/security/exec\_attr :- This file is responsible for maintaining

binary executable file info.

# vi /etc/security/exec\_attr ↵

foto last line

Profmgmt: Solaris: Cmd::: /usr/sbin/format: uid=0

↓  
Profile name      (or)  
                    SUser  
                    "Keyword"

:wq ↵

Role

## Creating a Role User :-

# roleadd : to Create Roll User.

# roleadd -d /usr/role1 -m role1 ↴

# Passwd role1 ↴

New : xxx

Confirm : xxx

# rolemod -P Profmgmt role1 ↴

## Dedicating Role to a normal User :-

# usermod -R role1 Sun ↴  
↓  
existing  
User name

as a Normal User to know how Many roles are available for him

\$ roles ↴

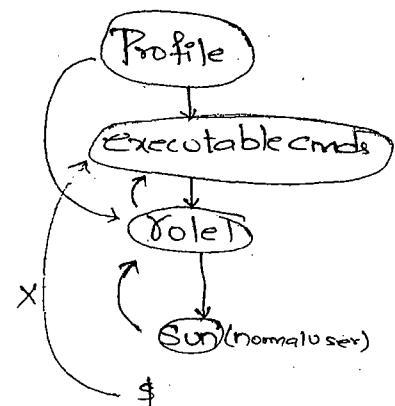
O/P: role1

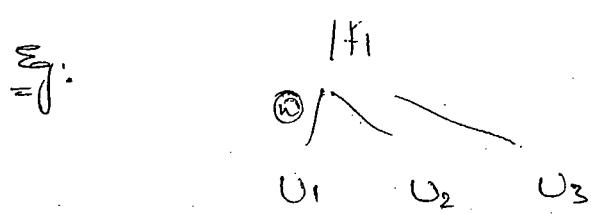
\$ su - role1 : to login as role User.

## Setfacl :- (Acl's Access Control list)

- file | directory level security.

- by using Setfacl And we can dedicate a special type of Permission to a Particular User.





If I give 777 permission to file then all users can access it but by using Setfacl cmd we can give write permission to user  $U_1$  only.

- If a file has 777 Permissions also we cannot remove it from ~~normal~~ User level.
  - If that file has writeable Permission at root level then we can remove. but by default it has 755  
 $\text{drwx}|\text{r-x}|\text{r-x}$ 
    - change permission & then we can remove.

# Set fact -m U: Sun: &w\*, m: &w- If 1  
↓ ↓ ↓  
Modify User Mask value  
name (or)  
effective level.

#getfac | f1 ↴

We Can See The Setfacl Permissions.

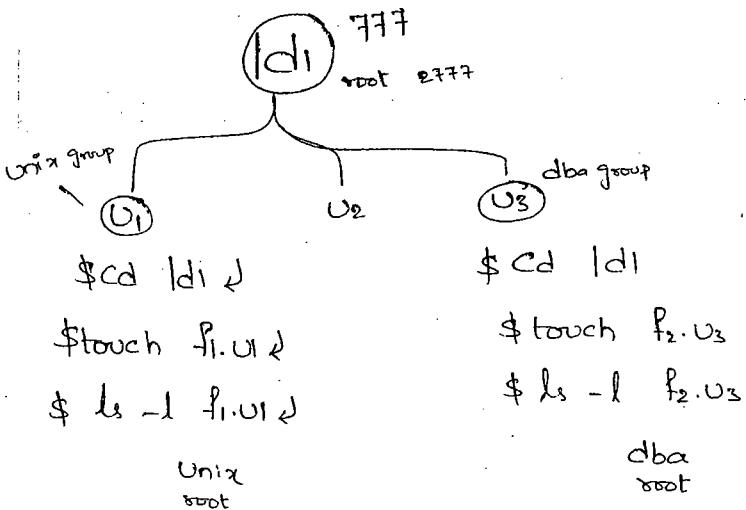
$U: \text{Sun} = \delta w x$ ,  $M: \gamma w -$

To delete Set fac1 Permissions

⑥ → User Sun has 810x now.

# Setfac -d U: sun: &wx, M: &w- Hi |

Setgid: to Create Common group for all the Users.



If we Specify a group to ld1 then Even though we Create files under U1/U2/U3 the group will be same group given to ld1.

-As a normal user we can directly Enter into /tmp directory & Create files. By default Setgid is available for /tmp.

Implementing Setgid :-

`id -a <username>`

=to know who has

#mkdir -m 777 ld1 &

#chmod 2777 ld1 & : numerical Way Setgid

#chmod g+s ld1 & : Symbolic Way Setgid

#ls -ld ld1 &  
(as) /tmp

drwxrws/rwx

disabling

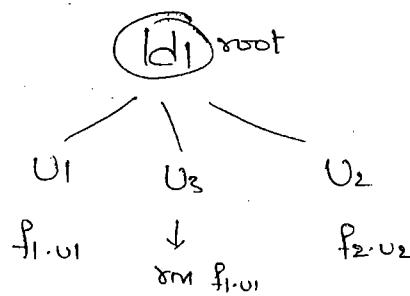
# chmod 777 ld1 &

# chmod g-s ld1 &

Drawback:

Any User Can Remove the file in root directory

of root group



- But by Using **Sticky bit** only owner can remove the file.

**Stickybit + Setgid** works.

## Implementing Sticky bit

# chmod +t ld1 ↴

# chmod 1777 ld1 ↴

4555 → Setuid

2777 → Setgid

1777 → Stickybit

**Setgid + Stickbit**

# chmod 3777 ld1 ↴

## Demo:

- Create Some <sup>(3)</sup> group & User accounts

# listusers ↴

## Setuid demo:

# telnet 0

\$ format ↴

\$ /usr/bin/format ↴

exit

```
# Which format  
# ls -l /usr/sbin/format
```

```
# chmod u+s /usr/sbin/format
```

```
# chmod 4555 /usr/sbin/format(as)
```

```
# ls -l /usr/sbin/format.
```

```
# telnet 0  
:babu
```

```
$ /usr/sbin/format ↵
```

```
exit
```

```
# telnet 0  
:sun
```

```
$ /usr/sbin/format ↵
```

drawback: Any user can execute format

```
# chmod u-s /usr/sbin/format
```

Rbar's demo :-

```
# vi /etc/security/Root-attr ↵
```

Last line

Profngt :::

:wq ↵

```
#vi /etc/security/exec-attr ↵
```

Last line

Profngt : Solaris : Cmd :: : /usr/sbin/format : uid=0

:wq ↵

Creating a new user.

```
# roleadd -d /usr/role1 -m role1  
# Password role1 ↴
```

Dedicating a Profile to a new User.

```
# rolemod -P Profmgmt role1 ↴
```

```
# usermod -R role1 sun ↴
```

```
# roles sun ↴ : to check roles of Sun user.
```

```
# telnet o ↴
```

```
Login: sun ↴
```

```
$ roles ↴
```

```
$ /usr/sbin/format ↴
```

```
$ su - role1 ↴
```

```
Password: ↴
```

```
$ /usr/sbin/format ↴
```

Su ↴

taking Complete authorisation of user home dir.

/etc/user-attr:

Maintains Profiles & roles info.

setfacl demo:

```
# cat file ↴
```

```
# ls -l file
```

```
# chmod 777 file
```

```
# ls -l file
```

Create a file

```
# Setfacl -m u:sun:rwx,m:rw- f1 ↴
```

```
# ls -l f1
```

-rwx-r--r-- +

+ → indicates Setfacl Perm.

```
# getfacl f1 ↴
```

```
# telnet o ↴
```

Login: Sun ↴

Pwd:

\$ Pwd

\$ cat f1

\$ cat >> f1

Ctrl+d

exit

```
# telnet o ↴
```

Login: babu

Pwd:

\$ cat f1 ↴

\$ cat >> f1 ↴

No Permission.

If we want to give Privileges to all group members.

```
# Setfacl -m u:sun:rwx:g:unix:rw-,m:rw- f1
```

26/IV

## Web Server's :-

Apache :- Apache is a Webserver which is used to put an organization information intranet. (or) Internet

- We have different types of Webserver's available in Market.

Eg:- Apache, Tomcat, iplanet, weblogic etc.,

- Compared to any other Webserver Apache is very fast.

- Apache is having only one Main Configuration file i.e., **httpd.conf** the location is

**/etc/apache/httpd.conf**

- Easy to Configure & Easy to administrate.

- Daemon is: **httpd**

- Port number is: **80**

## Packages for Apache:

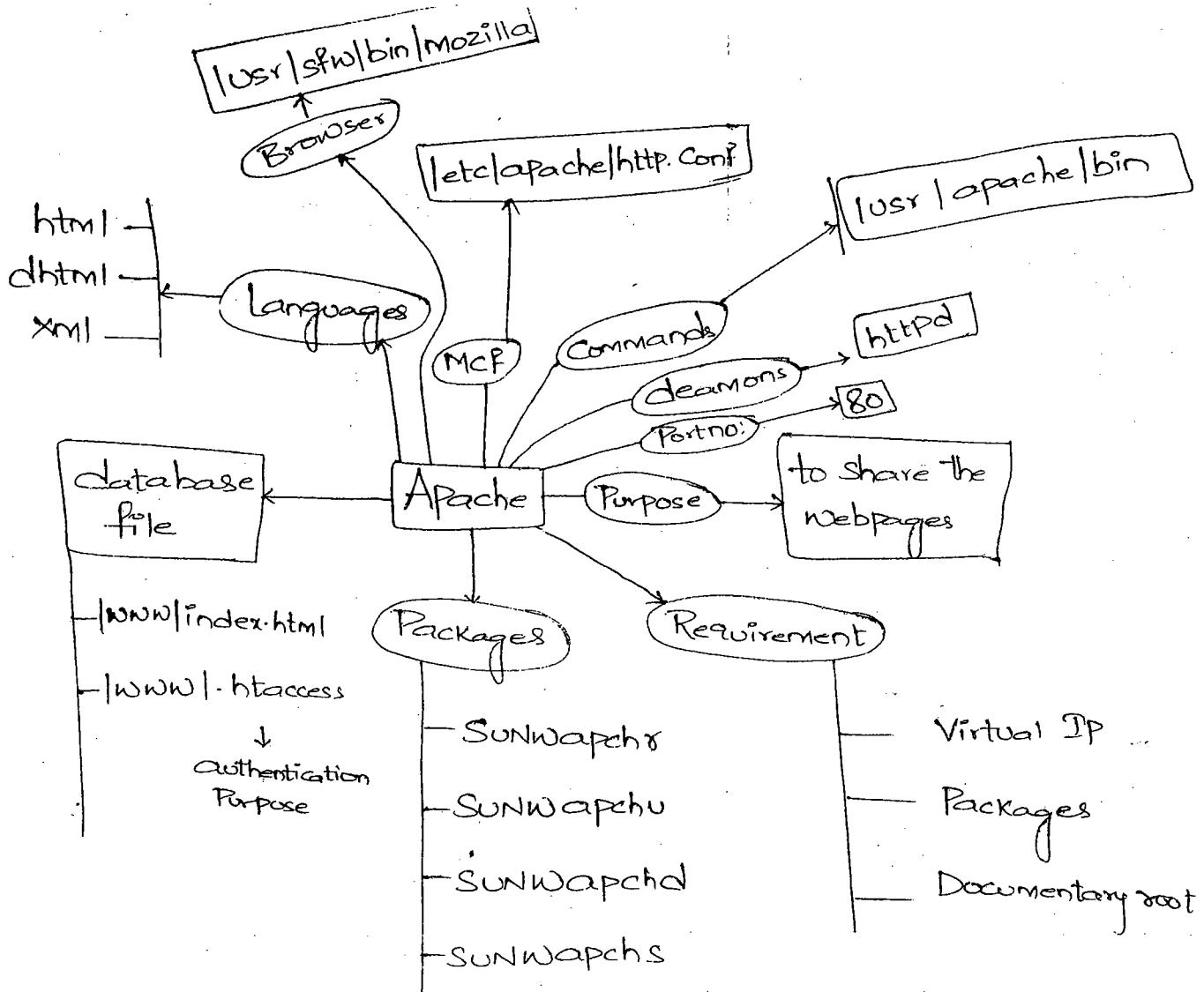
- All Pkgs are available under 4/4 cd

\* SunWapache

\* SUNWapache

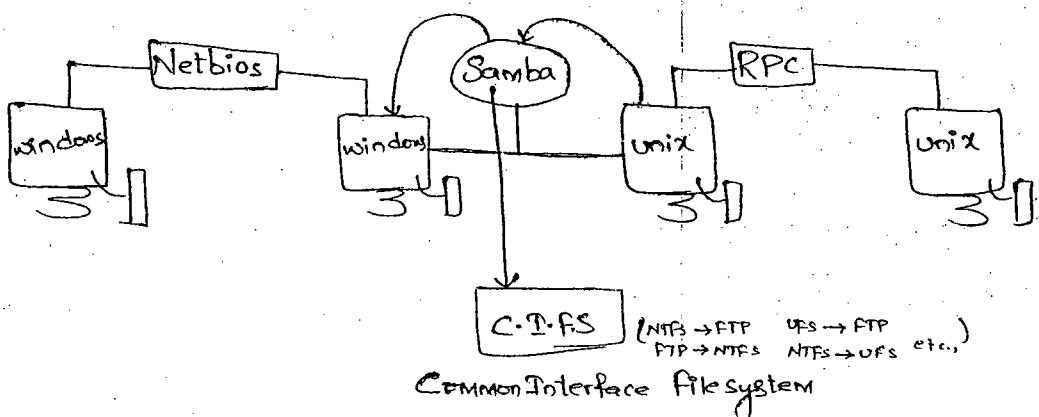
\* SUNWapchd

\* SUNWapchs



## Samba Server:

- Samba is a file Server.
- Samba is used to Share the files & directories.  
from Windows → unix, Unix → Windows etc.



## Packages:

- \* SUNW Smbau
- \* SUNW Smbac
- \* SUNW Smbad
- \* SUNW Smbas

## Main Configuration file:

[etc | Sfw | Smb. Conf]

Daemons: Smbd — (137) [Send message block daemon]  
Nmbd — (138) [net bieu block daemon]

Smbd: - is Responsible for updating Share folders.

Nmbd: - is Responsible for laying communication b/w  
Windows → Unix & Unix → Windows.

- Even we can <sup>Provide</sup> authentication also called  
Samba authentication Server.

- Samba works with Work group Concept.

## Command location:

[usr | Sfw | bin]

## Demo:

Insert H/H cd:

```
# cd /cdrom/cdrom0/Solaris-10/Product
```

```
# ls | grep SUNWapache (root environment)
```

SUNWapache (user " ")

SUNWapache1 (documentation)

SUNWapacheS (to make apache server)

```
# Yes; Pkgadd -d . SUNWapache SUNWapache SUNWapache1
```

SUNWapacheS ↳

Then we install the packages automatically 2 locations  
are getting updated that are etc/apache

cd etc/apache:

- find Main Config. file info.

~~etc~~ ↳ usr/apache

~~cd~~ cd usr/apache/bin:

- apache Related Cmds.

cd etc/apache ↳

# ls ↳

```
# cp httpd.conf-example httpd.conf ↳
```

```
# vi httpd.conf ↳
```

httpd : hyper text transfer Protocol

goto line 367

ServerAdmin root@westsunq

## Lab Demo: TCP-Wrappers

Purpose: We Can Deny Particular System Services

(  
ftp  
telnet  
nfs)

# Vi /etc/hosts.deny ↴

in.telnetd: 200.200.0.1

<hostname> in.ftpd

:wq ↴

# Svcadm restart inetd ↴

# inetadm -P ↴ : to check status true/false.

# inetadm -M tcp\_wrappers=true ↴

# Vi /etc/default/login

\* Remove Comment

not a Pst.

x # Console = login

# telnet 200.200.0.1

Login:

Passwd:

## Setgid Demo

```
# mkdir ldi
```

```
# ls -ld ldi
```

```
# mkdir -m 777 ldi
```

```
# chmod 666 2777 ldi ↴
```

```
# ls -ld ldi
```

```
# chmod g+s ldi ↴
```

```
# telnet o
```

login: sun

```
$ cd ldi ↴
```

```
$ touch f1 ↴
```

exit

```
# telnet o
```

login: u1

```
$ cd ldi
```

```
$ touch f2
```

```
$ ls -l
```

```
$ rm f1 ↴
```

```
$ exit ↴
```

Sticky bit: only owner can remove files.

```
# chmod 1777 d1 ↴
```

$g+s+t$

att

```
chmod 3777 ldi
```

```
ls -l ldi
```

```
ls -ld ldi
```

④ → Sticky bit

assign sticky bit & see the file.

goto line (58)

ServerName 200.200.0.9  
(Primary IP)

goto line (1053)

Copy the 7 lines & Paste at end of file.

② 7yy

goto line (1339)

Uncomment all these entries.

<Virtual Host 200.200.0.189 :80>

(Manually we need to create this virtual IP)

line (1340)

ServerAdmin root@wstsung

line (1341)

DocumentRoot /www

-We can customize documentary root also. (Manually we need to create)

Line (1342)

ServerName 200.200.0.9

:wq!

Create Virtual IP

```
#ifConfig -a
#IfConfig iprbo addif 200.200.0.189 up
```

To make Permanent

```
# cat >/etc/hostname.iprbo:1
```

200.200.0.189

ctrl+d

139

```
# vi /etc/hosts
```

Last line

```
200.200.0.189 wostsun189
```

:wq!

```
# mkdir /www
```

```
# cd /www
```

```
# cat >index.html
```

hai this is wilshiresoft.com

Ctrl+d

```
# cd /usr/apache/bin
```

```
# ./apachectl configtest
```

(Control script used to monitor MCF &  
Start daemons)

```
# ./apachectl Stop
```

```
# ./apachectl Start
```

```
# Pgrep httpd
```

Open Browser

```
# cd /usr/sbin/bin
```

```
# ./mozilla &
```

How to Provide Authentication for Web sever.

```
# CP /etc/apache/httpd.conf-example /etc/apache/httpd.conf
```

```
# vi /etc/httpd.conf
```

④ ServerAdmin root@wostsun9

④ ServerName 200.200.0.9

(HOM)

AllowOverride AuthConfig

(1053)

7yy Copy 7 lines.

Paste at end of file.

(1339)

<VirtualHost 200.200.0.188:80>

(1340)

ServerAdmin root@wstsun9

(1341)

DocumentRoot /www

(1342)

ServerName 200.200.0.9

:wq ↵

# ifconfig -a

# ifconfig eth0 addif 200.200.0.188 up ↵

# cat >/etc/hostname.ip00 :1 ↵

200.200.0.188

# vi /etc/hosts

200.200.0.188

:wq ↵

# cd /www

# cat > index.html

# vi .htaccess

AuthType Basic

AuthName "Logging"

AuthUserFile /password/pass

: Manually we need to Create this dir.

Require User tom jerry

now ↴

# Mkdir | Password

# Cd | Password

# Useradd -d /usr/tom -m tom ↴

# Useradd -d /usr/jerry -m jerry ↴

Create Passwords.

# Cd /usr/apache/bin ↴

+ ./apachectl Configtest ↴

# ./htpasswd -C /password/pass tom ↴

↓  
Create  
Pass file  
under  
Password directory

# ./htpasswd /password/pass jerry ↴

+ cat /password/pass. ↴

# ./apachectl stop ↴

# ./apachectl start ↴

# Pgrep httpd ↴

# ./mozilla & ↴

## Setgid Demo

```
# mkdir d1  
# ls -ld d1  
# mkdir -m 777 d1  
# chmod 777 2777 d1 ↵  
# ls -ld d1  
# chmod g+s d1 ↵  
  
# telnet o  
Login: sun  
$ cd d1 ↵  
$ touch f1 ↵  
exit  
# telnet o  
Login: u1  
$ cd d1  
$ touch f2  
$ ls -l  
$ rm f1 ↵  
$ exit ↵
```

Sticky bit: only owner can remove files.

```
# chmod 1777 d1 ↵
```

<sup>g+s+t</sup>  
att

```
chmod 3777 d1
```

```
ls -l d1
```

```
ls -ld d1
```

④ → Sticky bit

assign sticky bit & see the file.

## Lab Demo: TCP-Wrappers:

Purpose: We Can deny Particular System Services

(  
ftp  
telnet  
nfs)

# Vi /etc/hosts.deny ↴

in.telnetd:200.200.0.1

<hostname> in.ftpd

:W7 ↴

# Svcadm restart inetd ↴

# inetadm -P ↴ : to check status true/false.

# inetadm -M tcp\_wrappers=true ↴

# Vi /etc/default/login

\*Remove Comment not a Pw.

x# Console = login

# telnet 200.200.0.1

Login:

Password:

26/IV

## Web Server's:-

Apache :- Apache is a Webserver which is used to popularize an organization information intranet. (a) Internet.

- We have different types of Webserver's available in Market.

Eg:- Apache, Tomcat, iplanet, weblogic etc.

- Compared to any other Webserver Apache runs very fast.

- Apache is having only one main Configuration file i.e., **httpd.conf** the location is

**/etc/apache/httpd.conf**

- easy to Configure & Easy to administrate.

- Daemon is: **httpd**

- Portnumber is: **80**

## Packages for Apache:

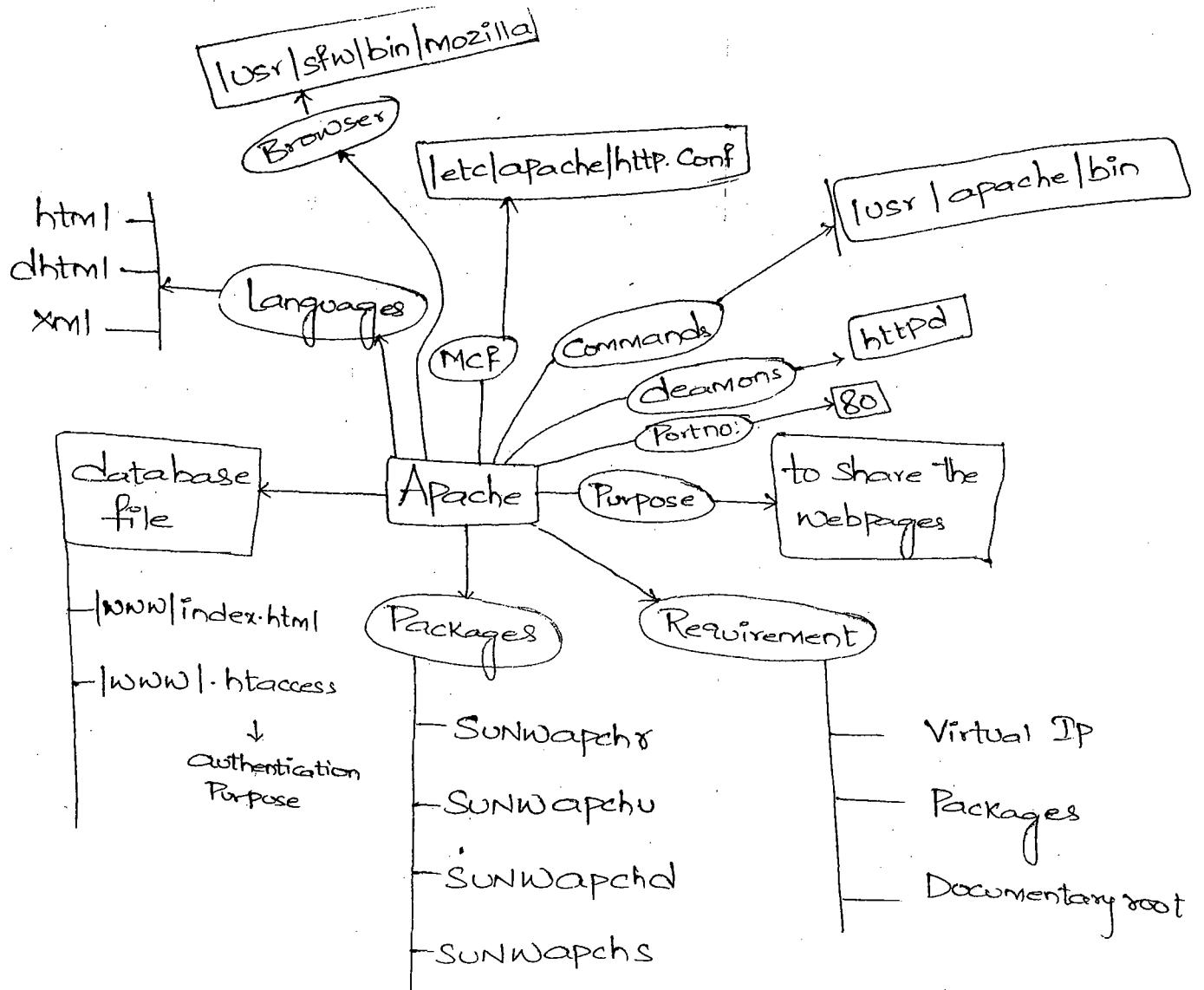
- All Pkgs are available under /etc/cd

\* SUNWapcr

\* SUNWapchu

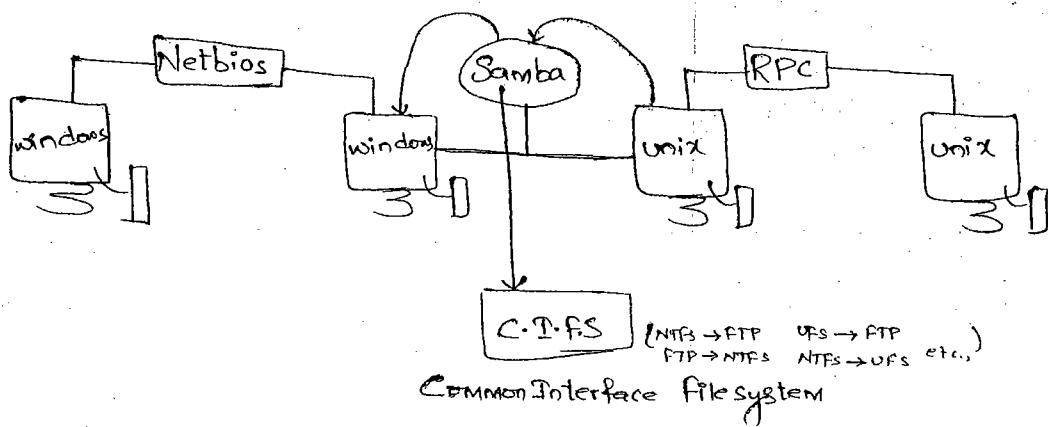
\* SUNWapchd

\* SUNWapchS



## Samba Server:

- Samba is a file server.
- Samba is used to Share the files & directories.  
from Windows → Unix, Unix → Windows etc.



```

# zonecfg -z Myzone
> add fs : used to add filesystem to nonglobal
    zone.
fb> Set dev=/dev/hdsk/c0t0d0s6 add ?
> Set Special dev=/dev/hdsk/c0t0d0s6
> Set dir=/newfs. : it automatically creates a directory
    in nonglobal zone.
> Set options=logging
> Set type=ufs
> end
> Verify
> Commit
> exit

```

for affecting the changes we need to reboot nonglobal zone

```
#zoneadm -z Myzone reboot : to reboot the ngz being
    in gz.
```

```
# zlogin -c Myzone
```

```
# df -h
```

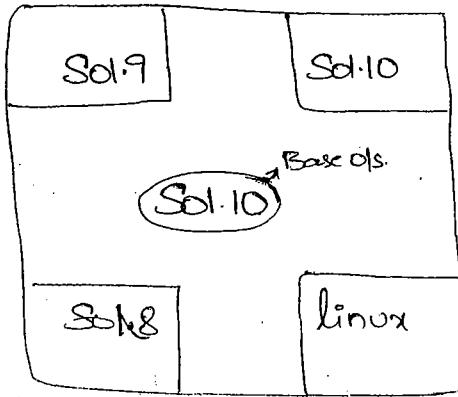
How to remove / modifying existing attributes in NGZ : being in GZ

```
# zonecfg -z Myzone
```

```
> Remove net physical=xtlso1 : if we click tab two times
    it displays possible options.
```

```
> info
```

```
> add net
```



## Demo : Branded Zones

Adding fs to NGZ from GZ :

In GZ

# format ↴

# !

> P

> P

1gb fs.

:9

# newfs

# zonecfg -z Sunzone

> add fs

> Set raw=/dev/rdsk/c0t1d0s0

Set special=/dev/dsk/c0t1d0s0

Set dir=/newfs

> Verify

> Commit

To restrict the Normal  
Users to login Create  
nologin file in /etc.

## Packages:

- \* SUNW Smbau
- \* SUNW Smbac
- \* SUNW smbad
- \* SUNW Smbas

## Main Configuration file:

[etc | Sfw | Smb. Conf]

Deamons:      [Smbd] — (137) [Send message block deamon].  
                        [Nmbd] — (138) [net bieu block deamon]

Smbd: - is Responsible for updating Share folders.

Nmbd: - is Responsible for laying Communication b/w  
Windows → Unix & Unix → Windows.

→ Even we can <sup>Provide</sup> authentication also called  
Samba authentication Server.

→ Samba Works with Work group Concept.

## Command location:

[usr | Sfw | bin]

## Dem0:

Insert 4/4 cd:

```
# Cd | cdrom | cdromo | Solaris -10 | Product
# ls | grep SUNWapch* (root environment)
    SUNWapchu (user *)
    SUNWapchd (documentation)
    SUNWapchS (to make apache server)
```

```
# Yes | Pkgadd -d . SUNWapch* SUNWapchu SUNWapchd
                                         SUNWapchS ↳
```

Then we install the packages automatically 2 locations  
are getting updated that are etc|apache

Cd etc|apache:

- find Main Config. file info.

~~etc~~ usr|apache

~~cd~~ cd usr|apache|bin:

- apache related Cmds.

(Cd etc|apache ↳

# ls ↳

# cp httpd.conf-example httpd.conf ↳

# vi httpd.conf ↳

goto line 567

ServerAdmin root@wstsung

httpd : hyper text transfer Protocol

goto line (108)

ServerName 200.200.0.9  
(Primary IP)

goto line (1053)

Copy the 7 lines & Paste at end of file.

④ 7yy

goto line (1339)

Uncomment all these entries.

<Virtual Host 200.200.0.189 :80>

↓  
(Manually we need to create this virtual IP)

line (1340)

ServerAdmin root@wstsung

line (1341)

DocumentRoot /www

- we can customize documentary root also. (Manually we need to create)

Line (1342)

ServerName 200.200.0.9

:wq ↵

Create Virtual IP

# ifconfig -a

# ifconfig iprb0 addif 200.200.0.189 up ↵

To Make Permanent

# cat > /etc/hostname.iprb0:1

200.200.0.187

ctrl+d ↵

```
# vi /etc/hosts
```

lastline

```
200.200.0.189 wstsun189
```

:wq!

```
# mkdir /www
```

```
# cd /www
```

```
# cat >index.html
```

hai this is wilshiresoft.com

Ctrl+d

```
# cd /usr/apache/bin
```

```
# ./apachectl configtest
```

: (Control script used to monitor MCF &  
Start daemons)

```
# ./apachectl stop
```

```
# ./apachectl start
```

```
# Pgrep httpd
```

Open Browser

```
# cd /usr/sfw/bin
```

```
# ./mozilla &
```

How to Provide Authentication for Web sever.

```
# CP /etc/httpd/apache2/httpd.conf-example /etc/apache2/httpd.conf
```

```
# Vi /etc/httpd.conf
```

(367) ServerAdmin root@wstsung

(385) ServerName 200.200.0.9

(HOM)

AllowOverride AuthConfig

(1053)

7yy Copy 7 lines.

Paste At End of file.

(1339)

<VirtualHost 200.200.0.188:80>

(1340)

ServerAdmin root@hostsun9

(1341)

DocumentRoot /www

(1342)

ServerName 200.200.0.9

:wq ↵

# ifconfig -a

# ifconfig ip8bo addif 200.200.0.188 up ↵

# cat >/etc/hostname.ip8bo :1 ↵

200.200.0.188

# vi /etc/hosts

200.200.0.188

:wq ↵

# cd /www

# cat > index.html

# vi .htaccess

AuthType Basic

AuthName "Logging"

AuthUserFile /password/pass

146

: Manually we need to create this dir.

Require User tom jerry

WPA ↴

# Mkdir | Password

# Cd | Password

# Useradd -d /usr/tom -m tom ↴

# Useradd -d /usr/jerry -m jerry ↴

Create Passwords.

# Cd /usr/apache/bin ↴

+ ./apachectl configtest ↴

# ./htpasswd -c /password/pass tom ↴  
↓  
Create  
Pass file  
under  
Password directory

# ./htpasswd /password/pass jerry ↴

+ cat /password/pass ↴

# ./apachectl stop ↴

# ./apachectl start ↴

# Pgrep httpd ↴

# ./mozilla & ↴

## DEMO. Ximics ver... .

Insert &/& cd

```
# cd /cdrom/cdromo/Solaris-10/Product/
```

```
# ls | grep SUNW$mbas
```

```
# yes | Pkgadd -d SUNW$mbas
```

```
# Pkginfo | grep SUNW$mbas
```

C  
d  
u

When we install this pkg it automatically updates \$local

```
cd /etc/sfw
```

```
cd /usr/sfw : Samba related cmd.
```

```
# cd /etc/sfw/
```

```
# ls
```

```
# cp Smb.conf-example Smb.conf
```

```
# vi Smb.conf
```

Line ②1

Workgroup = SPARC (anyname)

Line ⑦1

; → Commented

remove ;

Password Server = 200.200.0.9

Line 264

Copy 8 lines & Paste at end of file.

274

Uncomment

[Oracle] → manually Create this Particular directory.

Comment = -----

Path = /oracle

Valid Users = Smbaul Smbau2 (manually we need to create these users)  
Public = no

:wq

#mkdir /oracle

# useradd -d /usr/Smbaul -m Smbaul

# useradd -d /usr/Smbau2 -m Smbau2

assign Password's.

# cd /usr/sfw/bin

# ./Smbpasswd -a Smbaul

New Password:

Retype new Smb Password:

[etc|sfw|Private|SmbPasswd]

- Maintains Samba users Encrypted Password Info.

# ./Smbpasswd -a Smbau2

# ./Smbclient -U Smbaul -L localhost (Confirmation user is adding or not).

If errors

# cd /etc/init.d

# ./Samba Stop

# ./Samba Start

# cd /usr/sfw/bin

# ./Smbclient -U Smbaul -L localhost (info abt samba user)

Client Side Config. (^86)

- goto My Computer

Rightclick → Properties → Select Network Identification → Properties

Workgroup : SPARC

Click **OK** ↴

Reboot the windows Machine.

**Start** → Run

|| 200.200.0.9

Connects: Smbaul  
Password

Create a file under Smbaul.

Start Run.

telnet 200.200.0.9

27/10/09

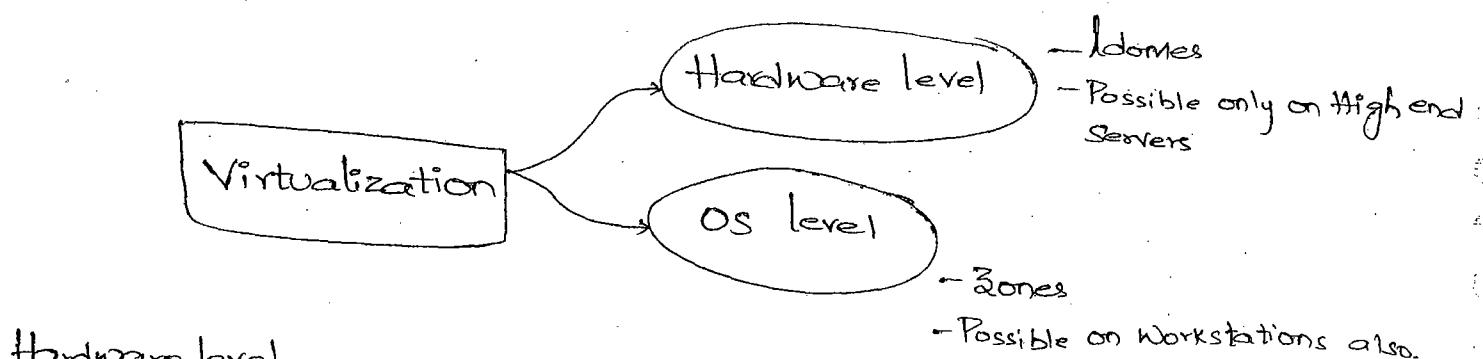
# ZONES

Zone's: (Virtualization)

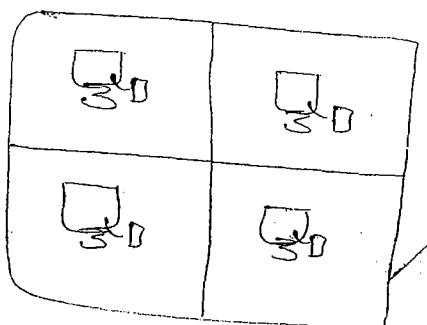
- Similar to VMWare.
- in built in Sun Sparc Machines.
- Entire Zones we can call as OS level Virtualization.
- base level OS is called Global zone

- Zones is an advanced feature in Solaris 10.
- Zones is a Concept of Creating Multiple Instances on top of a global Zone.

By Implementing the Zones we can come over Hardware level Virtualization.



hardware level



- Zones are Mainly Used for Checking Purpose.
- We can allocate Hardware Resources to two or More Zones
- The total administration could not be done on a non-global Zone.
- Zones provide virtual O/S Services that look like different Solaris instances to users & users applications.
- Each and Every non-global zones are maintain a special type of file system i.e., **LOFS** (loop back file system)

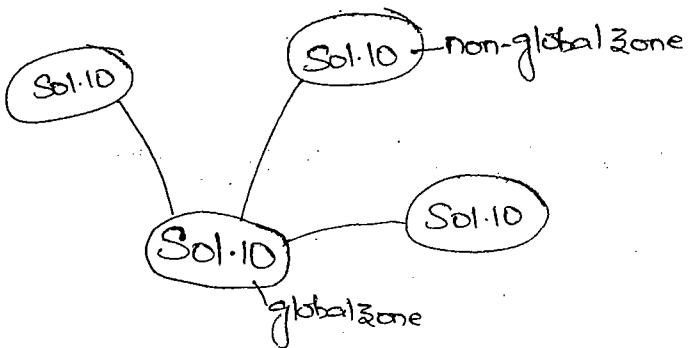
### Packages for Zones :-

- \* **SUNWZoneR** → to update the Root Environment.
- \* **SUNWZoneU** → to update User Environment.

- Each and Every Zone will act as an individual Machine like Virtual Operating System.

### features of Zones :-

1. Isolation
2. Security
3. Transparency
4. Application Installation.

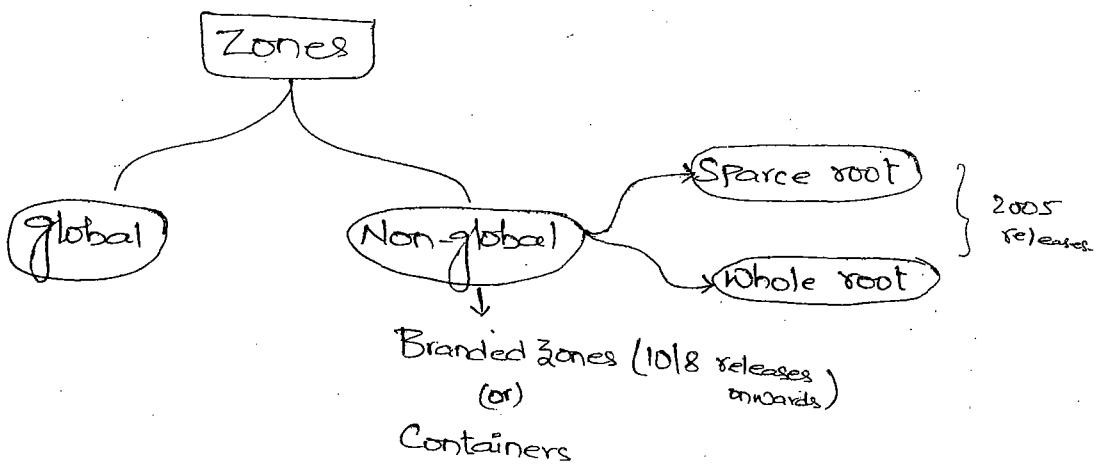


- Entire Non-global zones administration is done on global zone.

### Types of Zones :-

Zones is two types:

1. Global zone.
2. Non-global zone



### I. Global Zone :-

- It is a base OS.
- Complete administration can be done on Global Zone only.
- Global Zone id's are static.
- Always Global Zone id is '0'.

`#fzonename` is a command used to display zone name.

Op: Global

- Global Zone is used to Configure, Install, halt, Shutdown, Reboot, Uninstall, UnConfigure & etc.,
- Kernel Threads are been Shared from global zone only.
- It Provides the single bootable instance of the Solaris OS that runs on the System.
- All global zone Packages automatically inherited to non global zones.

### Non-Global Zone :-

- A non global Zone is created from global zone & Managed by it.
- At Max. We Can Create 8192 Non-global zones on a one Single Os.
- The Non global zone id's are not Constant. When we Reboot the Non global zone the id's are automatically changed.
- It Contains a subset of installed packages.
- It is not aware of the existence of other zones.

- Non global zones are divided into 2 types

### 1. Sparse Root Zone:

- Min. 300mb is needed for creating Sparse Root Zone.

- While installing the Sparse Root Zone it automatically inherit Packages from Global Zone.

### 2. Whole Root Zone:

- Min 4 to 5gb Space is needed for Creating Whole Root Zone.

- It won't maintain inherited packages from global zone but it copies all the Packages from global zone nothing but complete individual machine.

### Daemons for Zones:-

\* `Zoneadm`

\* `Zshed`

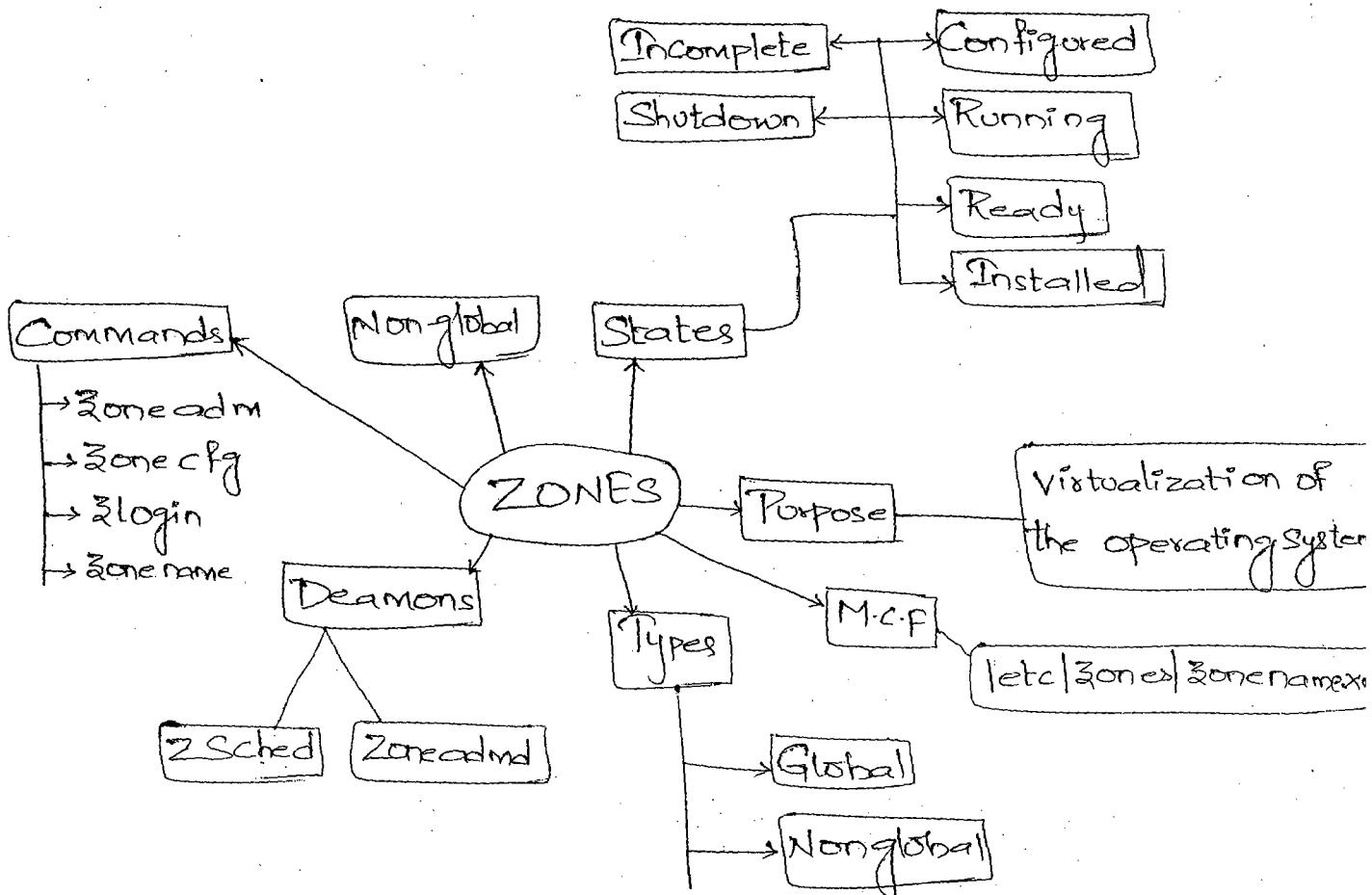
\* Zoneadm :- - Main Configuration daemon for zones.

- This daemon is responsible for allocating the zone id's & Start the Process. Called Zshed Process.

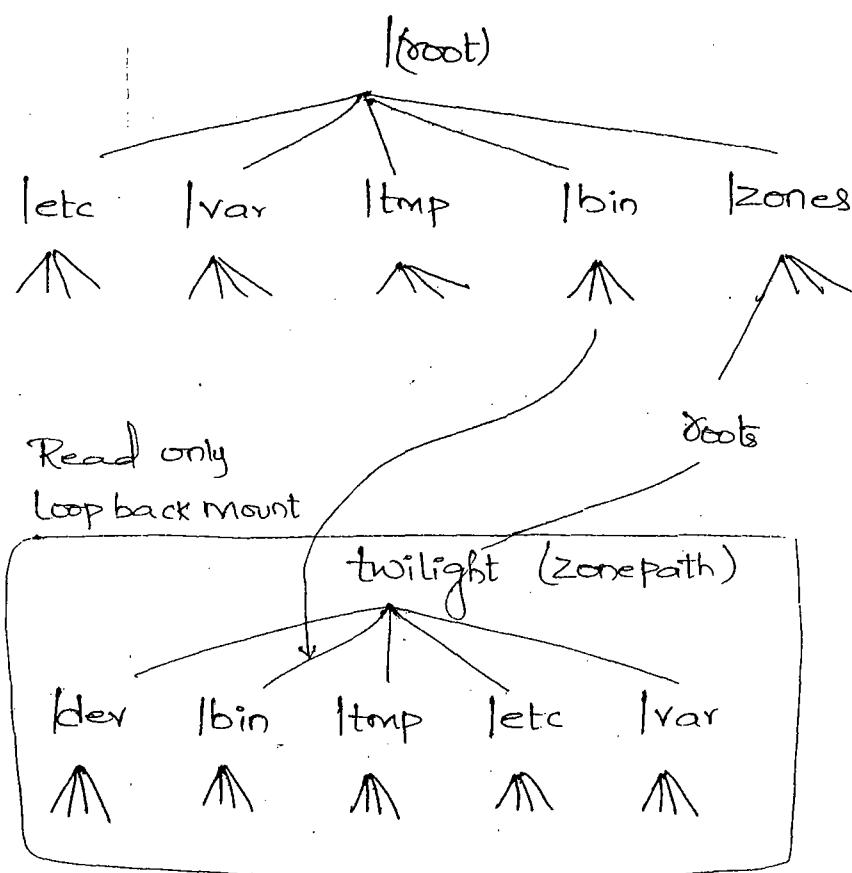
- Always Zoneadm deamon will try to start up.
- It Monitors Each & Every non global zone.
- It Mounts any loop back file Systems.

ZShed :- - It is a Scheduler deamon for all non-global zones.

- It Maintain Kernel Threads for non-global zones.
- It is always depends upon Zoneadm deamon.



## Global Zone



## Demo for Zones:-

(Previously Vxer)

```
# Pkginfo | grep SUNWzoner ↵
```

```
# Pkginfo | grep SUNWZoneu ↵
```

```
# Pkginfo -l SUNWzoner : displays full info.
```

# Create a file System & Mount it.

Note: we have to change the Mount point Permissions to 700.

```
# chmod 700 /myzone ↵
```

Creating Sparse Root Zone.

# zonename ↴

# zonecfg -z Myzone ↴

Zonecfg:myzone > Create ↴

Zonecfg:myzone > Set autoboot = true ↴

> Set zonepath=/myzone ↴

> add net ↴

Create -b : to create whole zone

Create : Sparse root zone.

autoboot=false : remains in obples

(adding nfo config. Info).

Zonecfg:myzone:net > Set Physical = rtls0 ↴

> Set address=200.200.0.91 ↴ : Specify Virtual IP.

(no need to create manually, it automatic plumb).

> info ↴

> end ↴

Zonecfg:Myzone > Verify ↴

: Checking the Configuration Errors.

> Commit ↴

: Used to Save the Config Info. Permanent.

> exit ↴

# Zoneadm list -vc ↴

: to Know Configured (or) not

After Configuring we need to install it.

# Zoneadm -z Myzone install ↴ : to install the zone Pkgs.

# Zoneadm list -vc ↴

: to check status

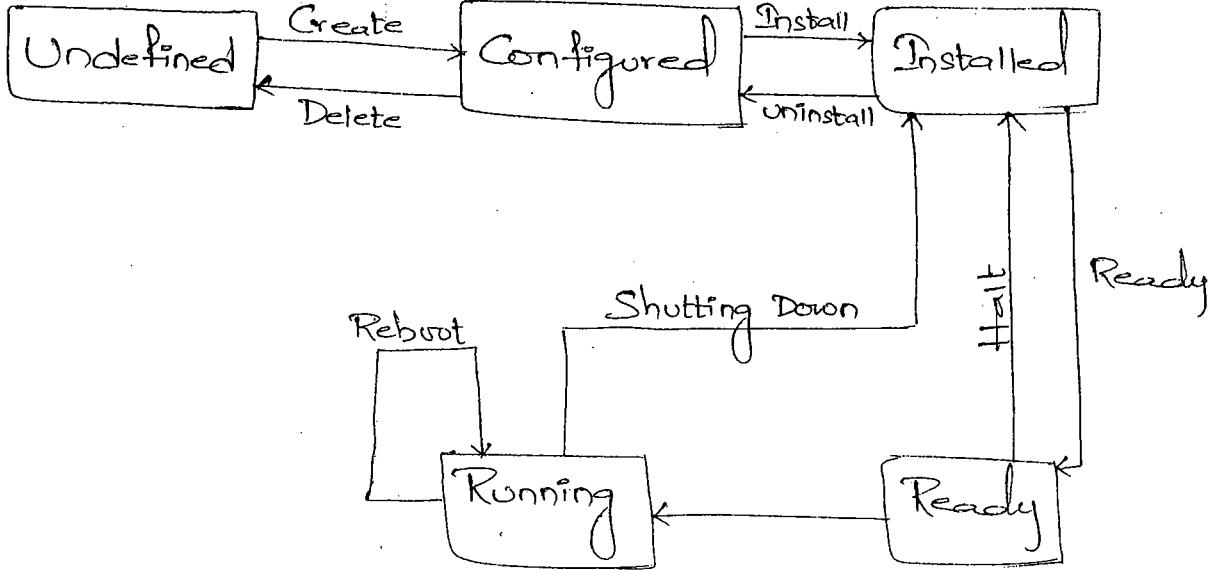
# Zoneadm -z Myzone boot ↴

: to boot zone

# Zoneadm list -vc ↴

: running status

## Zone States.



- When the Zone Status is Running then only it has id's.

Logging to nonglobal zone.

# zlogin -C Myzone ↴ (← for console)

Type Console no: : 13 ↴ (Common desktop environment)

Hostname :Myzone ↴

Kerberos Security : No ↴

Name Services : None ↴

My Console login: root ↴

# df -T ↴ {to check file system type.}

If we want to logout from Non-global zone.

# w.

(or)

# Ctrl+d

to login again

zlogin <Zonename>

How to change <sup>break</sup>  
Passwd. for zone.

add newfs.

### How to Remove Non-global Zone:

# zoneadm list -vc

If you want to delete a nonglobal zone

# zoneadm -z myzone halt ↴

# zoneadm list -vc

# zoneadm -z myzone uninstall ↴

# zoneadm list -vc ↴

# zonecfg -z myzone delete ↴

# zoneadm list -vc ↴

[etc/zones]

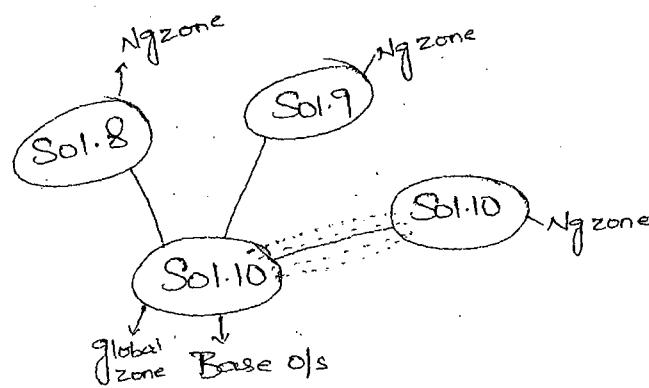
: Main Config. file of Zones.

8/10/01

## Branded Zones :- (Containers)

- Branded zones is a Concept of Installing Solaris 10

Previous releases on base OS.



Installing the Brandzone on Top of Sol.10 base OS

Packages for Sol.9 Container:

- \* SUNWbrandx
- \* SUNWbrandu
- \* SUNWbrandk

} These Pkg is used to install Sol.9 on top of Sol.10.

- We can Perform Sol.9 on Sparc Machines only. (in lab.)

- We can Perform Sol.9 on Sparc Machines only. (in lab.)

- If we want to install Containers on top of Sol.10 base OS

- If we want to install Containers on top of Sol.10 base OS  
We need to install "10/08" Solaris Version.

# zonecfg -z S9Brandd

Create -t SUNWSolaris9

> Create -t SUNWSolaris9

↓  
to specify  
template  
name

> Go with defaults.

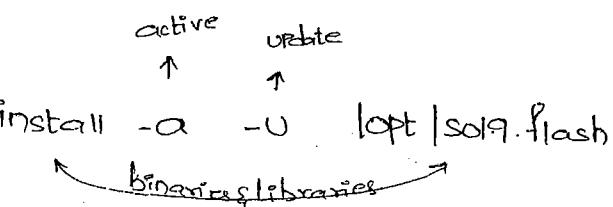
## Before

- Create "Sol.9 Flasharchive" on Solaris 9 only, ~~before~~ before installing
- Download the Sol.9 archive file from Sol.9 to Sol.10 O/S.

## To install

```
# zoneadm -z S9Brand
```

```
install -a -u lopt/sol9.flash
```



generally the binaries & libraries are taken from the base os when

We execute zoneadm -z S9Brand install. This is in normal zone

But we need to install binaries & libraries of Sol.9 OS in branded zones as the base OS is Sol.10.

## To boot

```
# zoneadm -z S9Brand boot
```

```
# zlogin -C S9Brand
```

## Installing Linux O/S on top of Sol.10 base OS:

(only on x86)

- Download the "CentOS" Linux Controller from the net.

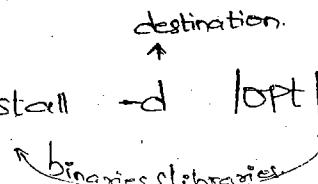
## While Configuring

```
# zonecfg -z linuxzone
>create -t SUNWbx
```

SUNWbx is template file for  
linux

## To install

```
# zoneadm -z linuxzone install -d lopt/Centos...
```



Then boot & login to linuxzone.

```
# uname -a
```

## Break Zone Password :-

```
# Prod  
# cd /myzone  
# cd root  
# vi etc/shadow  
remove 2nd field.  
:wq
```

```
# zonecfg -z Myzone info
```

: To Know what type of zone.

DIP: Inherited .....

then Sparse root Zone

otherwise

Whole root Zone

```
# zonecfg -z Myzone verify
```

: To Verify.

How to add a file System from global zone to  
Nonglobal Zone?

- We cannot create any filesystem in Non-global zone. But  
we can dedicate filesystem from global zone to nonglobal zone.

```
# format
```

Create a Partition & build the file system.

- don't Mount it.

# Vxassist

Build the -fis & Mount it.

### Creating Raid (I+O) :

# Vxassist -g oray make Str-mir 3oom layout=Stripe-mi  
Strwidth=64

# Vxassist

Build the -fis & Mount it.

### Resizing/it

To See license Keys

# Vxlicrep

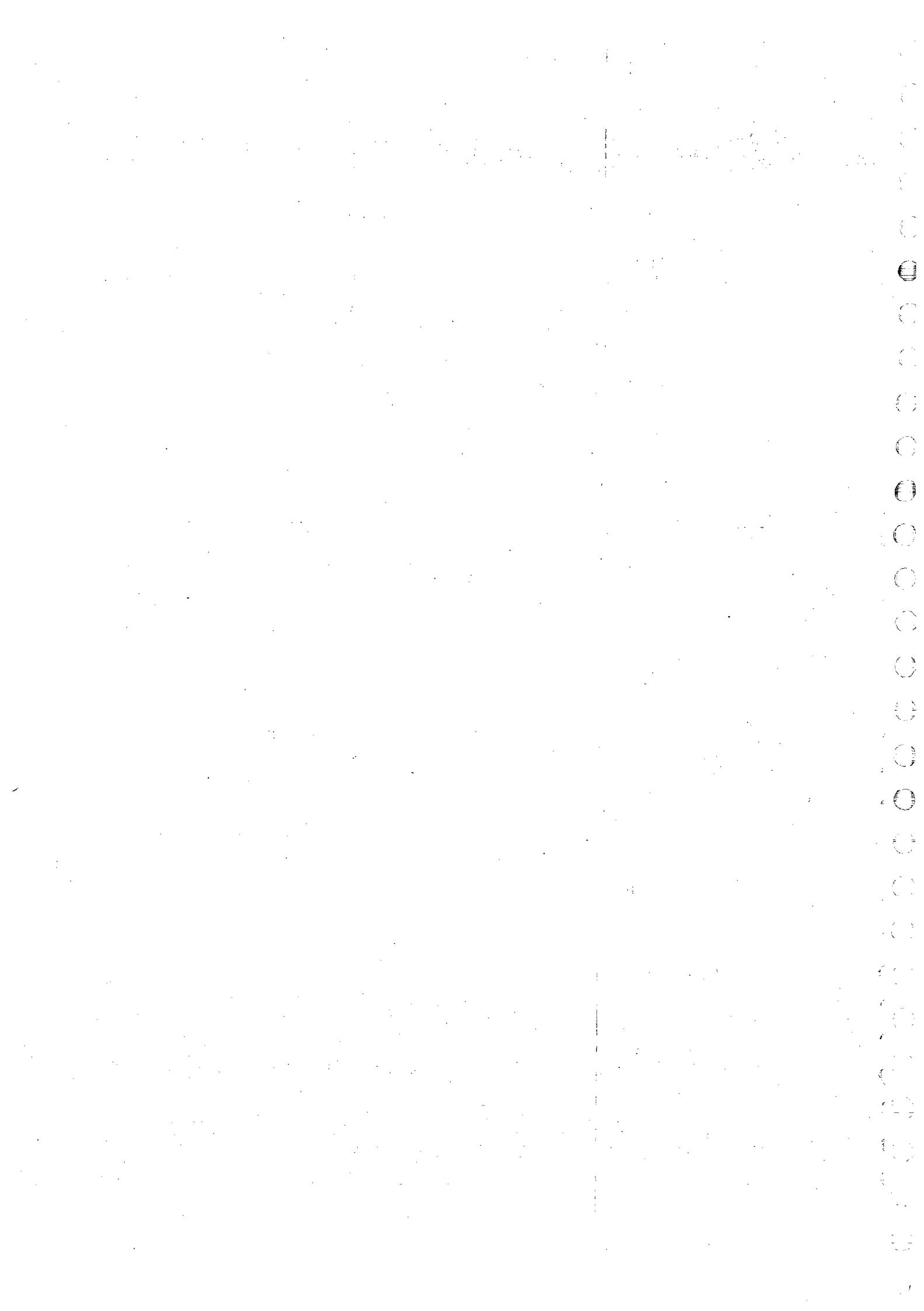
-displays license keys & also dummy  
license keys.

# Vxlicinst

: to change the license key.

- location of Mksa license keys

cd /etc/vx/licenses/lic



unstable

# zoneadm -z Sunzone reboot ↴

for Every Reboot Zoneid's will change.

We can remove -fs also.

zonectf -z Tajzone

> Remove fs dir=/nroot  
> info ↴

## Bridged Zones : Demo:

- Install 10/08 Release Version of Solaris.

### Demo for installing Sol.9 Content

Download Sol.9 flash archive file from Sol.9 OS.

after downloading

# cd /opt

# file Sol9.flash ↴

# cd

Download Container Pkgs from CJs Server:

# cd /CONTAINERS

# cd Containers

# ls ↴

# gunzip S9Contai.gz ↴

one file will generate tar archive file.

# tar -xvf S9contai ↴

We get a file S9Containers bundle.

```
cd sgContainers - bundle
```

```
# ls
```

```
# cd 1.0
```

```
# ls
```

```
# cd Product1
```

```
# ls : displays bz Packages.
```

(support brand utilities)

most

wsx

```
# Yes; Pkgadd SUNWsgbrandk SUNWsgbrandr SUNWsgbrandu ↵
```

```
# Pkginfo | grep SUNWsgbrandk ↵
```

8 ↵

0 ↵

```
# df -h
```

Create a file with 3gb & Mount it.

```
# chmod 700 /S9BRAND
```

```
# ls -ld /S9BRAND
```

```
# Zonecfg -z S9BRAND
```

```
# Create -t SUNWsolans9
```

```
# Set autoboot=True
```

```
# Set zonepath=/S9BRAND
```

```
# add net
```

```
# Set Physical=hwsl
```

> Set address= 200.200.0.169

> end

> Verify > Commit > exit

# zoneadm list -vc

Install

# zoneadm -z S9BRAND install -a -u /opt/Sol9.flash

# zoneadm list -vc

# zoneadm -z S9BRAND boot

# zoneadm list -vc

# zlogin -c S9BRAND

Language : 0

Locale :

Terminal : 7

Specify System Identification.

login:

# uname -v | a

## Installing the Linux Zone:

Install Sol.10/s release on top of X86 Machine.

```
# Cat /etc/release
```

Download the Centos\_fs\_image.tar.bz2 from CJs Server.

```
# Cd /opt
```

```
# ls
```

```
# file Centos_fs_image.tar.bz2
```

Create a file & Mount it.  
<sup>3gb</sup>

```
# Chmod 700 /linuxzone
```

### Configure:

```
# Zonecfg -z linuxzone <
```

```
> Create -t SUNWolv <
```

```
> Set autoboot=true <
```

```
> Set zon
```

```
> Set zonepath = /linuxzone
```

```
> add net
```

```
> Set Physical = el0000go
```

```
> Set address = 200.200.0.129
```

```
> end
```

```
> Verify
```

```
> Commit
```

```
> exit
```

#zoneadm list -vc

#ifconfig -a

### Install

# zoneadm -z linuxzone install -d /opt/Centos-fs-image  
tar.b2z

# Lab on Project: Secondary DNS

Config: Server 200.200.0.3

#domainname sun.com

# domainname > /etc/defaultdomain

# Svcs domain

# Svcadm restart domain

# Svcadm enable domain

# Mkdir -p /var/named

# cd /var/named

# ls

# vi named.hosts

:

|          |      |    |               |
|----------|------|----|---------------|
| hostname | IN A | IP | → server      |
| hostname | IN A | IP | → slaveServer |
| hostname | IN A | IP | → client      |

:wq!

# vi named.ver

:

3 IN PTR wstsun3  
 20 IN PTR wstsun20  
 17 IN PTR wstsun19

:wq!

vi /etc/resolv.conf

:

# Svcadm restart dns|server)

# Svcadm enable dns|server)

Config: client 200.200.0.19

#domainname sun.com

# domainname > /etc/defaultdomain

# Svcadm restart domain

# Svcadm enable domain

# cp /etc/nsswitch.dns /etc/nsswitch.conf

# vi /etc/nsswitch.conf

change 18th line  
dns <→ files

:wq!

#vi /etc/resolv.conf

domain sun.com

nameserver 200.200.0.3

nameserver 200.200.0.20

:wq!

# Svcadm restart dns|client

# Svcadm enable dns|client

# Svcs dns|client

# Svcs domain

# nslookup

>200.200.0.19

...

After Slave Config

#nslookup

200.200.0.19

----- SERVER -----

```
#domainname sun.com
# domainname >/etc/defaultdomain
#Svadm restart domain
#Svadm enable domain
#Mkdir -p Var/named/slave
# Vi /etc/named.conf
```

```
options {
    directory "Var/named/slave";
}

Zone "sun.com." {
    type slave;
    Masters { 200.200.0.3; };
    file "named.hosts";
}

Zone "0.200.200.in-addr.arpa" {
    type slave;
    Masters { 200.200.0.3; };
    file "named.rev";
}
```

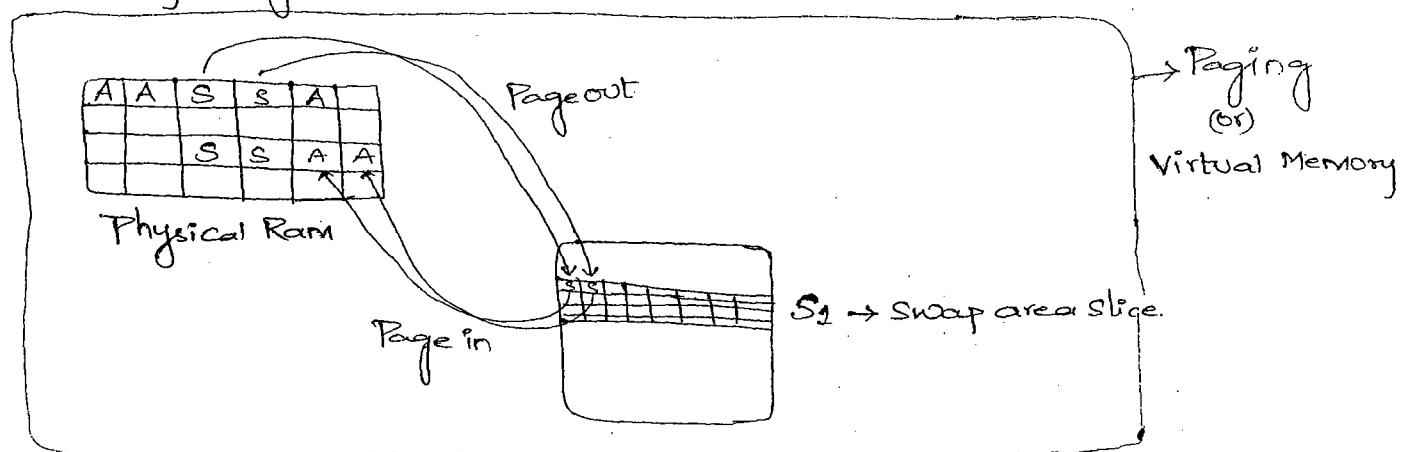
:now:

```
# Svadm restart dns/server
# Svadm enable dns/server
# Svcv dns/server
```

910101

## SWAP MANAGEMENT:-

- It's the regular task of the administrator dedicating an hard disk space to Swap area is called Swap Management.
- As an admin. regularly we need to monitor, adding (or) removing the Virtual Memory Statistics.
- Always Physical Ram can hold Active Process info.
- Always the Virtual Ram can hold Sleeping Process info.
- Each and Every Physical Ram's are having Registers, Responsible for maintain the Process info. in the form of Pages or Memory Pages.



Proc: Contains all Active Process info. in the form of directories.

Cd Proc

Procfs is the file system for Proc.

Swap Space we can dedicate in 2 ways

\* file system level

\* file level

\* file System level:

How to add Swap space in file system level.

# format ↴

Create a Partition.

Note: don't build the file system. (Don't run newfs cmd)

# Swap -a /dev/ldsk/c0t0d0S7 ↴

- used to dedicate Hard disk space to Swap area

# Swap -l ↴ : displays available Swap area

# Swap -s ↴ : display the combination of Physical RAM and Virtual Memory.

How to delete Swap Area?

# Swap -d /dev/ldsk/c0t0d0S7 ↴ : delete Swap area.

# Swap -l ↴

Vmstat (Virtual memory statistics.)

# vmstat ↴ : Used to display Virtual mem. Statistics.

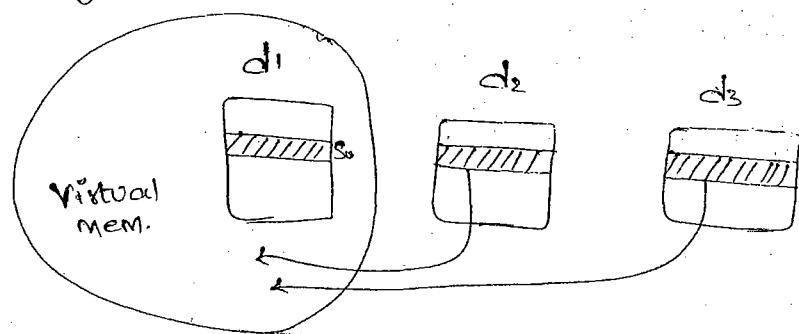
- As Admin generally we need to Identify Pageins & Pageouts

- The Combination Of Physical RAM + Memory & Swap Memory is called a "Virtual Memory".
- Whatever the Pages are transferred to from Physical RAM to Swap are is called "Page out's".
- The Pages transferring from Swap area to Physical RAM is "Page in's".
- SWAPFs is a file System Responsible for maintaining the Swap Area.
- By default one Memory Page size in Sparc Machine is 8K.
- By default one Memory Page size in X86 Machine is 4K.

`#Pagesize` ↳ : displays Memory Page Size.

`#PitConf -v | grep -i Mem` ↳ : displays Physical Memory Size <sup>(RAM)</sup>

- Swap file System Can Span across Multiple disks.



```
# Vmstat 1 10 ↴
```

-with in 1 min the dp will display 10 times.

How to add Swap Area by Creating a file?

\* file level:

```
# Mkfile 50M |Swapfile ↴  
↓  
Size of  
file.
```

when all Slices are filled and cylinders are not available then we can dedicate a file size to swap area. by creating file in root dir.

Adding Swap file:

```
#Swap -a |Swapfile ↴      after adding  
#ls -l |Swapfile ↴
```

-We are adding Swap file to Swap Spc.

-By default Sticky bit is enabled to Swap file.

Deleting Swap file:

```
#Swap -d |Swapfile ↴ : deletes Swap file.
```

If /tmp fils is full

(or)  
Swap fils is full then goto /tmp and execute #ls -l /tmp  
then check timestamp of files & remove unused files (or) most useful files.

(ltmp) → Universal dir.

to see all the users accessing ltmp directory

```
# fuser -u ltmp ↴
```

```
# fuser -k ltmp ↴
```

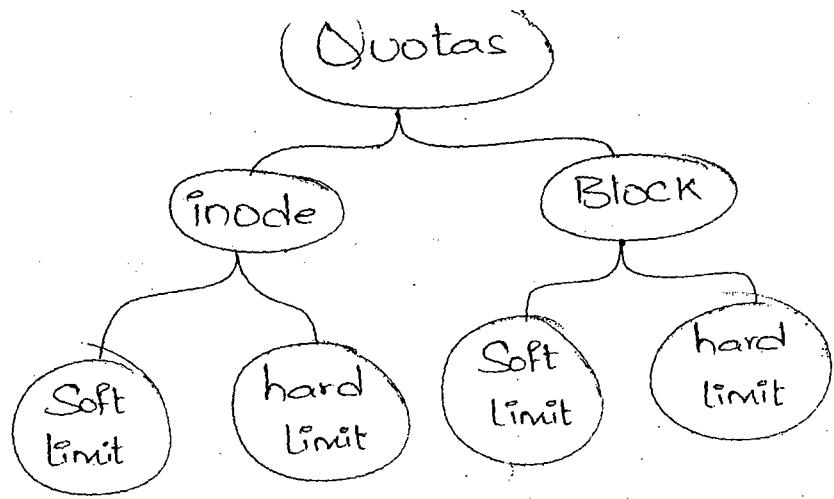
- to kill the users accessing the  
ltmp directory.

to kill a particular user.

```
# kill -9 <pid>
```

## Quotas:

- By implementing Quotas we can restrict users with limited space on hard disk.
- Always we have to implement Quotas on file system only.
- Quotas we can implement in two ways
  - (i) Inode level Restriction
  - (ii) Block level Restriction.



### Commands Related to Quotas:

- \* `quota on` → Turn on | enable Quota's
- \* `quota off` → Turn off
- \* `edquota` → edit quotas limit for normal user's
- \* `repquota` → used to display report for quotas

### Implementing Quota's:

- Demo Part.

# JOB Automation :-

Multiple Cmds at a time

# ls & Pwd & Ps &

Process :

Process is nothing but executing a Job at CPU level is called Process.

We have two different types of Processes

1. Foreground Process (fg)
2. Background Process (bg)

Regular background Process  
Automatic background Process

## 1. Foreground Process (fg)

- Cmds executing at \$ (or) # Prompt  
the cmd o/p we can identify directly on Monitor.

## 2. Background Process :

- It is Internal Process

Eg: daemons.

## Regular Background Process:-

- The Jobs which we are sending manually to background is called Regular Background Process.

E.g.: Sleep 100 & ↵

# cp f<sub>1</sub> f<sub>2</sub> & ↵

## Automatic Background Process :-

- This Process Which It Will Start Automatic When We Power on the System & Which it automatic Gets killed at System Shutdown time.

E.g.: daemons.

## Def'n Job Automation:

- A Process of executing a Particular job in a background with time Specification. In Regular Intervals is called Job Automation.

- We can Automate the Jobs in 2ways

→ at

→ Cron

\* "at" Jobs: These Jobs are Single time Executed Jobs.

- Each & Every "at" Job ends with an extension ".a".

E.g.: 112346.a

- We can Perform 3 actions in "at" Jobs Now Today Tomorrow

- The Main Config. file for "at" Job

etc/cron.d/at.deny

→ file is Responsible for Restricting Normal users for at Cmd. If?

- By default any user can perform at job.

`[at.allow]` ↴ : This file is responsible for allowing users to access "at" cmd.

If two file exists the priority will be for "at.allow" file only.

daemon for "at" jobs `Cron`

Log location for "at" Jobs:

`/var/spool/cron/atjobs`

→ all spooled / queued "at" jobs are located under this particular file.

Commands related to "at" Jobs:

`# at -l` } list spooled at jobs.  
`# atq` }

`# at -r <jobid>` ↴ : used to remove the at jobs.

`# tty` ↴ to know terminal type.

`# at now+1min` ↴

> banner will > /dev/lpt1/3  
> Ctrl+d ↴

} to specify the timestamp.  
after 1min banner cmd will execute on the given terminal.

## \* Cron :-

- Cron Jobs are used to execute a Job with regular intervals.
- These Jobs are we can execute multiple times.
- By default any users can execute Cron Jobs. (access)
- If you want to restrict any one of the user we have to execute:

# vi /etc/cron.d/Cron.allow ↴

<User>

:wq!

/etc/cron.d/Cron.allow

/etc/cron.d/Cron.deny

— Main config. files.

- All Spooled Cron Jobs are located under

/var/spool/cron/crontabs

## Cron :

\* Crontab -e → edit the Cron Jobs

\* Crontab -r → Remove the Cron Jobs.

\* Crontab -l → listing the CronJobs.

# Crontab -e ↴ : It will automatically open a text editor.

| ①<br>Minutes | ②<br>Hours | ③<br>day of the Month | ④<br>Month of<br>the<br>year | ⑤<br>day of<br>the<br>Week | ⑥<br>Script<br>file. |
|--------------|------------|-----------------------|------------------------------|----------------------------|----------------------|
| (0-59)       | (0-23)     | (1-31)                | (1-12)                       | (0-6)<br>↓<br>Sunday.      | ---                  |

→ → Specific days  
- → Specific range.

Ex: vi \$1 ↴

tar -cvvf /tar.bkp user

now ↴

# Crontab -e

30 18 \* \* \* Sh 1\$1

Demo for Swap:-

# Pagesize ↴

# PtfConf -v | grep -i Mem ↴ : displays Mem. Size.

adding fils to an existing Swap area:

# Swap -l

# format ↴

Create a Partition. Don't run newsfs.

```
#Swap -a /dev/dsk/c0d0s7
```

```
#swap -l ↴
```

```
#swap -s ↴
```

Edit the `/etc/vfstab` entries.

- After adding to make Permanent

```
# vi /etc/vfstab ↴
```

Lastline

```
dev/dsk/c0d0s7 - - Swap - yes - ↴
```

```
:wq ↴
```

Deleting a Swap Space:

Delete Entries `/etc/vfstab` file

```
#swap -l ↴
```

```
#swap -d /dev/dsk/c0d0s7 ↴
```

```
#swap -l ↴
```

When we delete ...

```
# vmstat 1 3 ↴
```

## Demo: Quotas

Create a file & Mount it

Create some user accounts under that file.

```
# useradd -d /users/Mark -m Mark ↴
```

```
# useradd -d /users/Clark -m Clark ↴
```

```
# cd /users.
```

```
# touch quotas ↴
```

```
# cd
```

```
# quotaon /users ↴
```

```
# clear
```

```
# edquota Mark
```

Text editor.

File /users block's (soft=0, hard=0) inodes (soft=3, hard=6)

Save & exit

```
# telnet 0
```

Login: Mark.

```
# cd
```

```
# touch f1 f2 f3
```

# repwota | users

### Remove Quotas:-

# quotaoff | users ↴  
# cd /users ↴  
# rm .quotas

# telnet o

:  
Now Create files. We can do.

### Demo: Job Automation :-

#### "at" Jobs:

# tty ↴  
# at now+1min  
at> banner Will > /dev/pts/5 ↴  
at> ctrl+d  
# cd /var/spool/cron  
# cd at/jobs  
# ls ↴

Users have restrictions.

## Cron jobs:

#Crontab -e

Open Text editor.

Come end of file.

\* \* \* \* \* ls -l > /dev/pts/5 <

Save & quit.

#Crontab -l

## Lab Demo:

### Root File System Backup

# cd /kernel/drv : Contains all loadable module info.  
(drivers).

# ~~mod~~ mt st ↴

# Ufsdump -cvf /dev/ram10 /kernel/drv ↴

# cd /kernel

# rm -r drv ↴

# init 6 ↴

—will goto Panic mode | OBP level

# ok> boot net -s

# Prod

# mkdir /ram

# cd /ram

# Mount /dev/dsk/c0t0d0s0 /tmp/root/ram

# df -h

# cd /ram

# ls

# cd /kernel

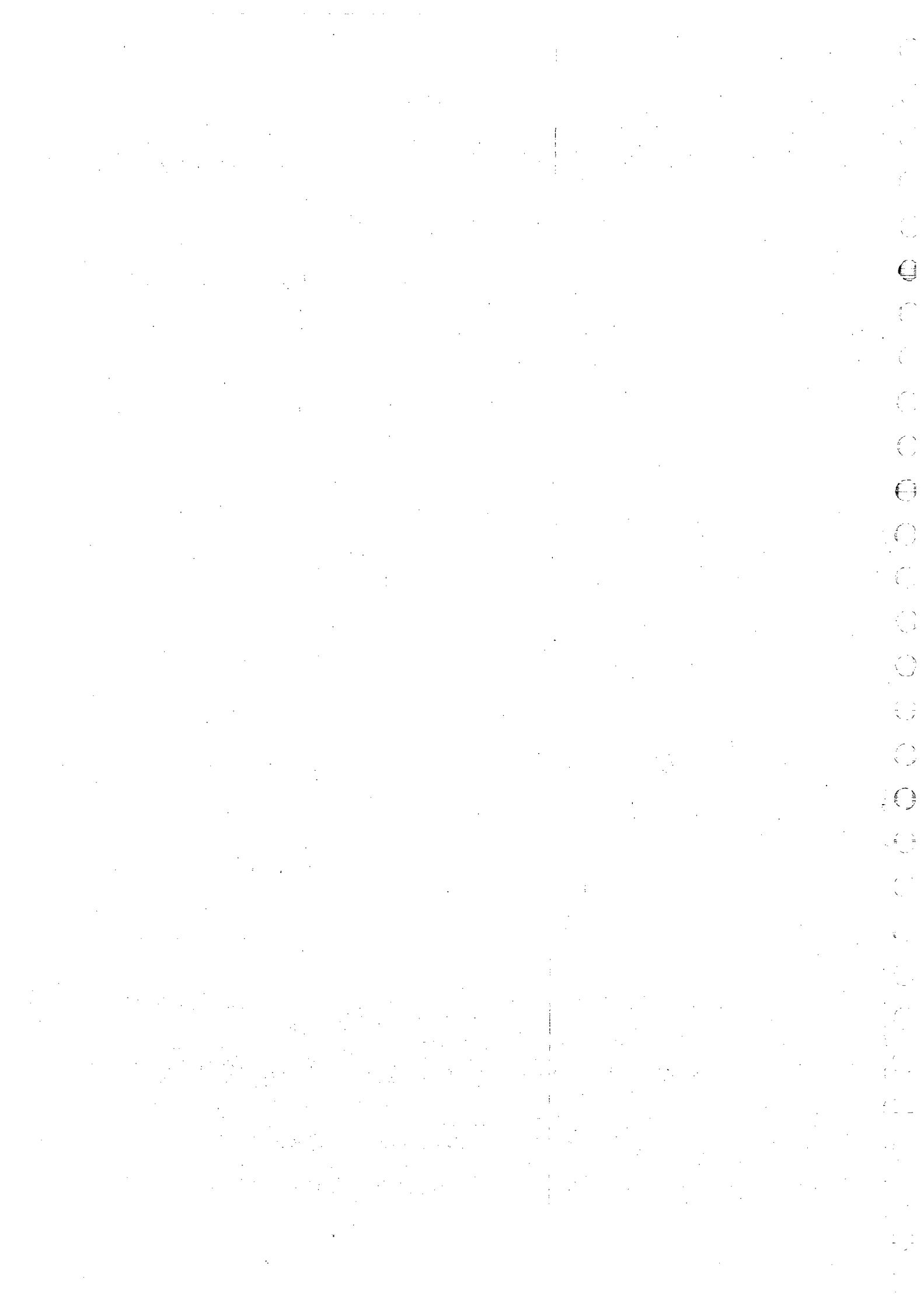
# cd ..

# mt st

# UFS restore -Xvf /dev/ram10 ~~/dev~~ /kernel ↴

# Specify volume: 1

# cd /kernel #ls



30/10/01

## SOLARIS VOLUME MANAGER

- It enables you to Create logical (or) Virtual Object.

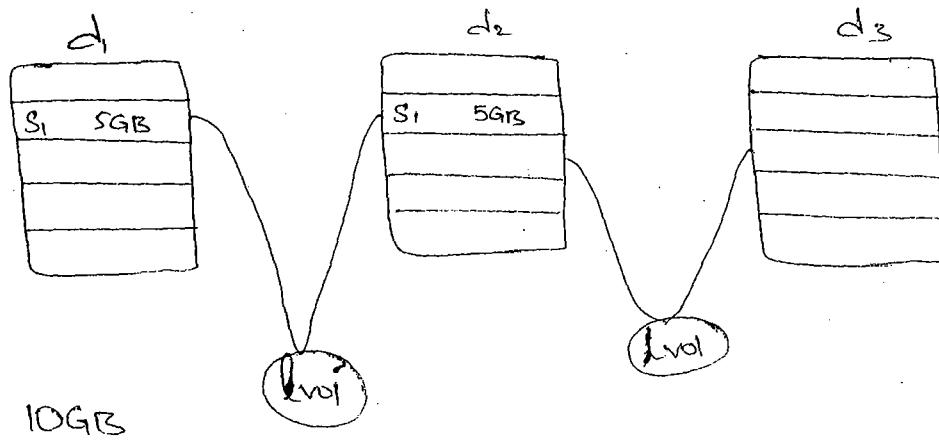
### Drawback of Traditional File System:-

1. A Single P/s Can't Span across Multiple disks
  2. Limited no: of Slices (only 0-7)
  3. We Cannot increase (or) decrease Filesystem Size's.
  4. Overlapping is Possible, it may Corrupt filesystem.
- 
5. We Can't implement RAID levels
  6. No Availability, No Redundancy.

### Advantages in Solaris Volume Manager:

- SVM Provides a huge Storage Environment by Combining 2 or More Slices.
- One Single file System Can span across multiple disks.
- There is no limitations of Slices (or) Partitions. We can create 'n' no: of Slices.
- We can increase (or) decrease file system sizes.

- We can implement RAID Levels.
- It provides Availability, Redundancy & High Performance.



- We can Configure & Manage SVM in two ways:

1. Command line Interface
2. Graphical User Interface.

— all the SVM Commands will start with "meta" in

C.L.I

Install  
Complete

— in sol. 10 Gui tool is `# smc &` ↴ (H14 cd)

`# admintool &` ↴ (sol. 9)

— SVM uses a special type of driver used to  
Configure the volumes. i.e., "Metadisk driver".

`# svcs -a | grep Meta`

— List SMF Services for SVM

## Terminology :-

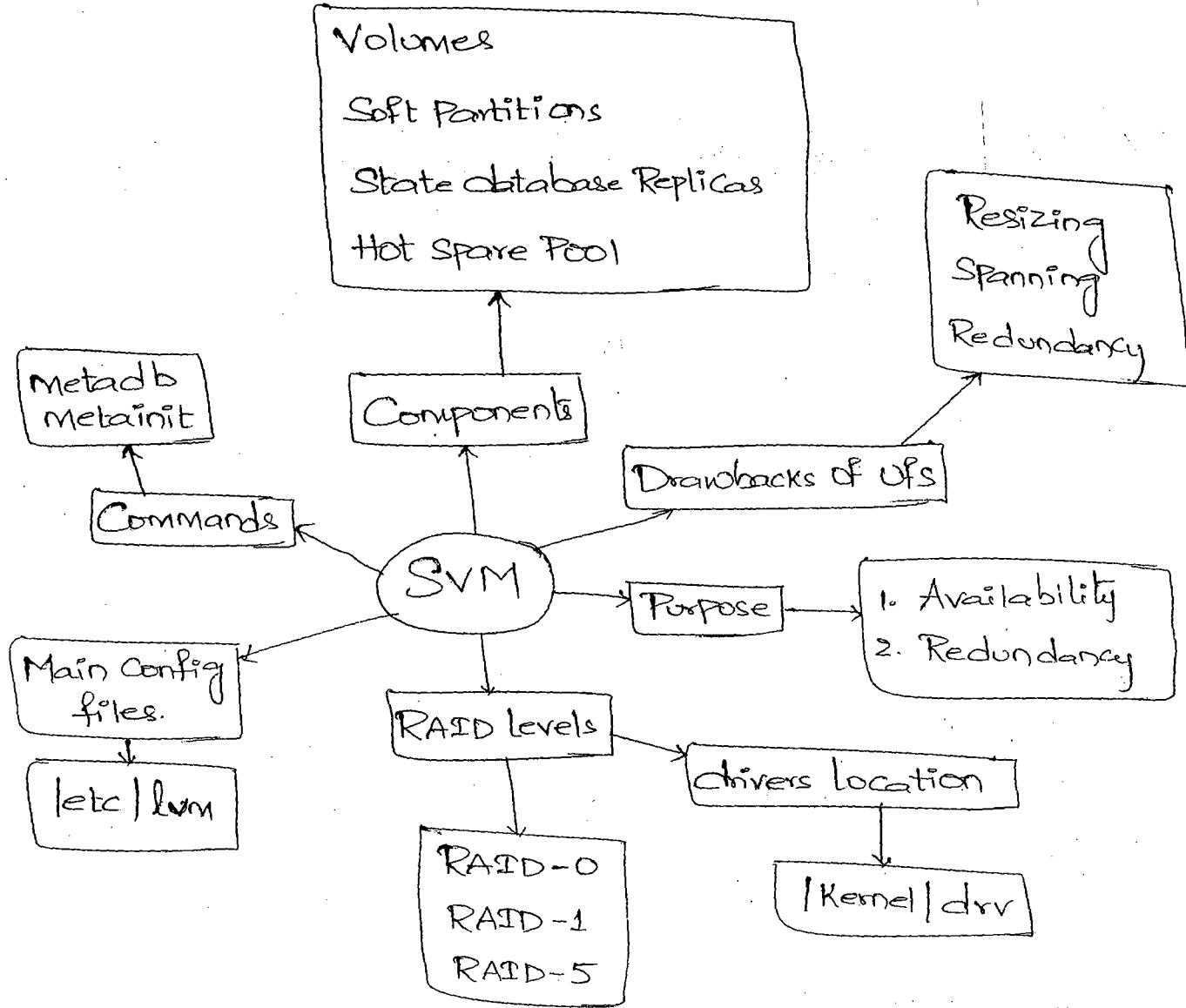
1. Metadatabase | state database
2. Volume
3. Hotspare Pool
4. Soft Partition.
5. RAID Levels.

## RAID Levels:

- RAID Stands for Redundant Array of Inexpensive disks.

- We have different types of RAID levels.

- (Non Redundant volumes.)
- \* Raid 0 → Concatination  
→ Striping
  - \* Raid 1 → Mirroring
  - \* Raid 5 → Data with Parity



## I. Meta Database | State Database

- Responsible for Maintaining Config.

Info. about all the logical volumes.

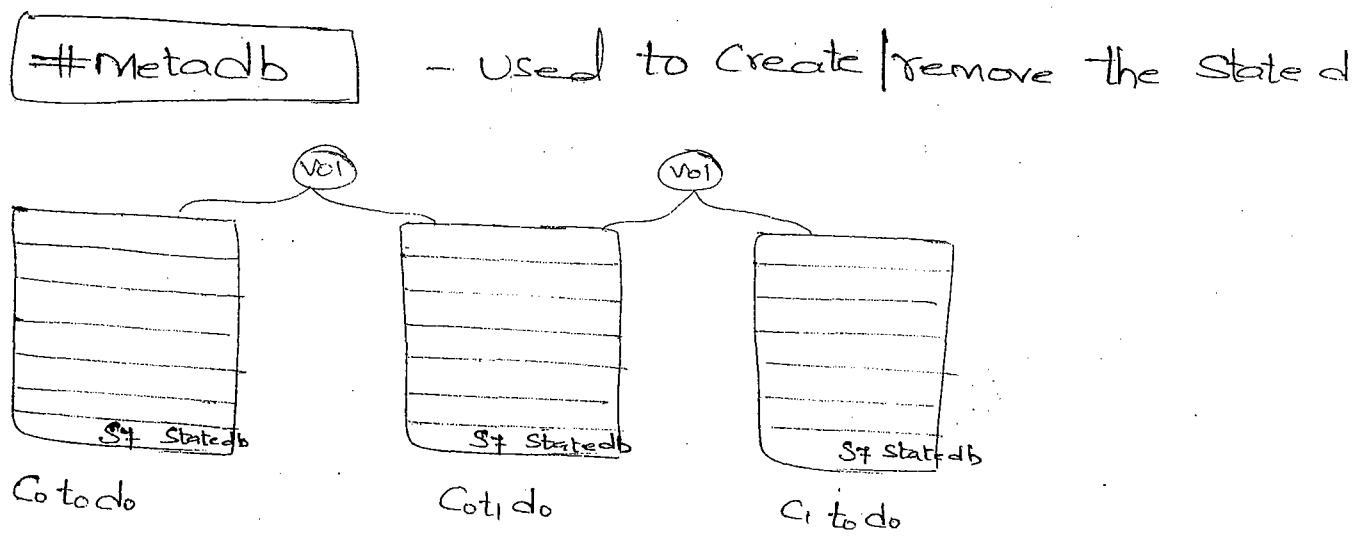
- It Maintains Volumes, Hot Spare

Pool, Soft Partition etc., Info.

- By default Meta database size

is "HMB".

- `/etc/lvm/Md.tab` → Maintain State db Predef tables info.
- `/etc/lvm/Md.db` → Maintain Configured State db info.
- It is Recommended to have Many Replica's of the Meta database on Multiple disks.
- We can define the State db. Status with the help of Controller level.



- We need to Create a State db on Each & Every disk Prior to Volume
- If any one of the State db is lost we can get the info. from another available State db.
- Without State db we can't create (or) Implement Logical Volumes.

## 2. Volume:

- Combination of 2 (or) More Physical Slices.
- One Single Volume Can Span across Multiple disks.
- We Can increase / decrease Volume Size dynamically.
- By default we can create 128 Volumes only.
- By editing one Parameter Even we can Create the logical volumes at Max. 8192.

# Cd |Kernel|drv

# Vi mdc.conf ↴

md = 128 <sup>Change</sup> ↴  
8192.

:wq ↴

We have to restart the "Meta" Services.

- The naming convention for volumes starting with 'd' character

"d #####"

## 3. Soft Partition:

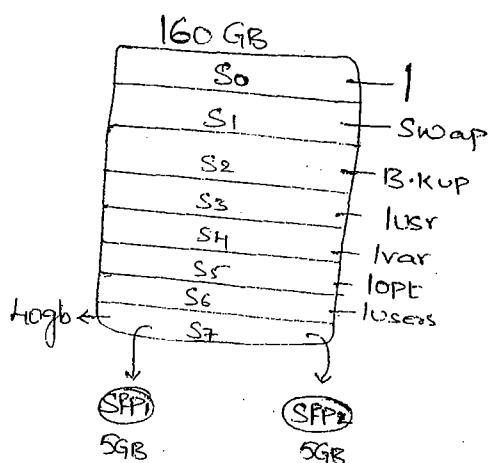
- It's a piece of portion in a slice is called soft partition.

- It's a special type of volume permits

the creation of very large storage devices.

- By implementation of the soft partitioning we can overcome the limitations of D-F Slices.
  - There is no limitation for creating soft Partition

## Volumes.



Don't run News in S7 slice  
only do specify Cylinders &  
Create Soft partitions.

#### 4. Hotspare Pool :-

- Collection of Multiple Physical Slices.
  - Hot Spares Provide increased data Availability RAID 1, RAID 5 Volumes.

Naming Convention Starts with

"hsp # # #"

## 5. RAID Levels :

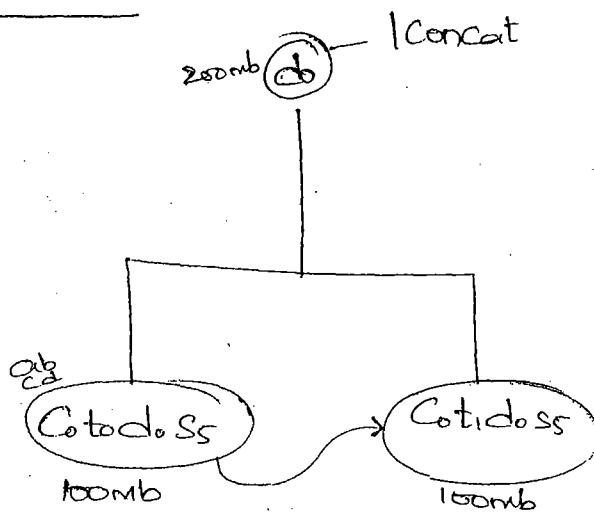
\* RAID 0 :- It's a non Redundant Raid level.

A hand-drawn diagram consisting of a horizontal line with a small vertical tick mark positioned exactly at its center.

## Striping.

# RAID Concatenation

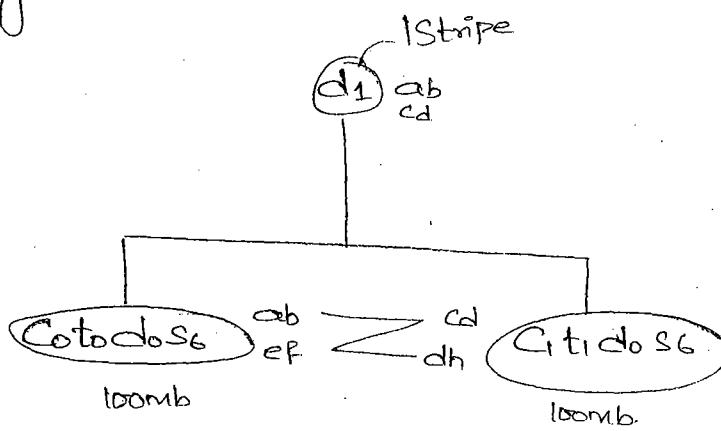
## Concatenation:



first the data is written  
on Cotidoss then on  
Cotidoss.

- In this Raid Technique there is no availability & low performance.
- If any one of the slice is corrupted we cannot get back the data.

## Striping :-

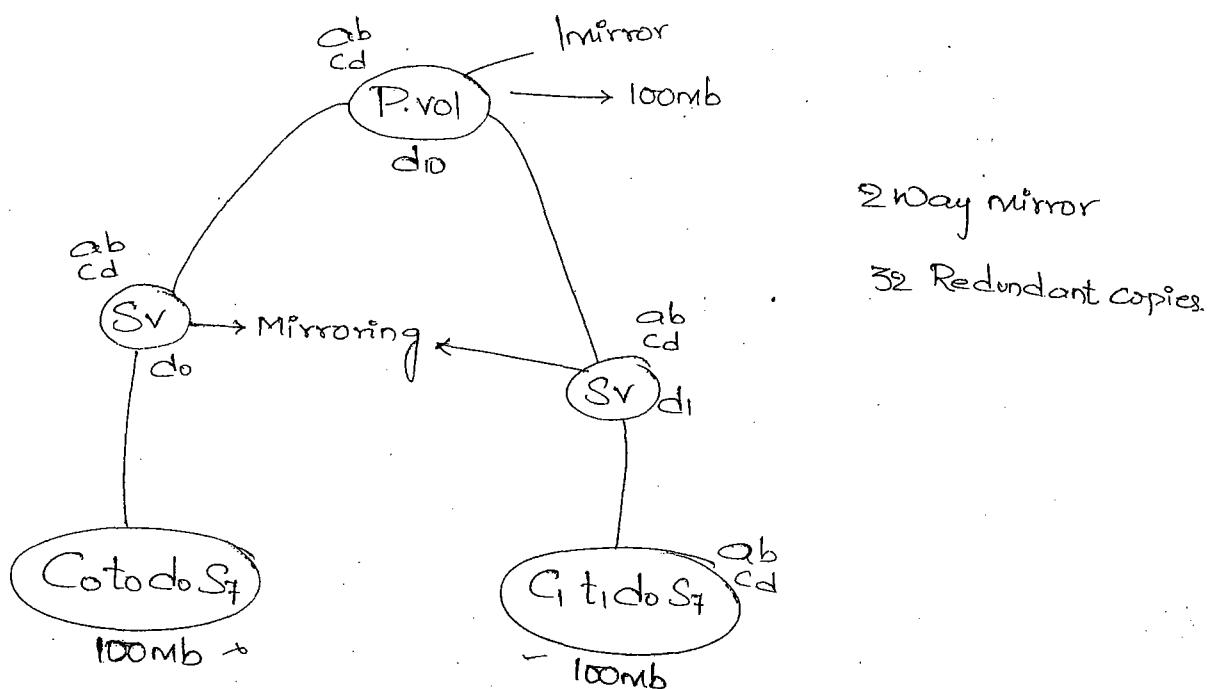


- No availability in case of any slice ~~is~~ corrupted.
- High Performance.
- The Physical Slices Sizes Should be ~~same~~ same.
- Based on Interlace value we can say which is concatenated & which is striped volume.
- By default Interlace value is 32kb.

- first 512 KB will be written on first disk & then remaining 32KB data will be written on another.

`#metastat` ↪ Used to display Metadevices Statist

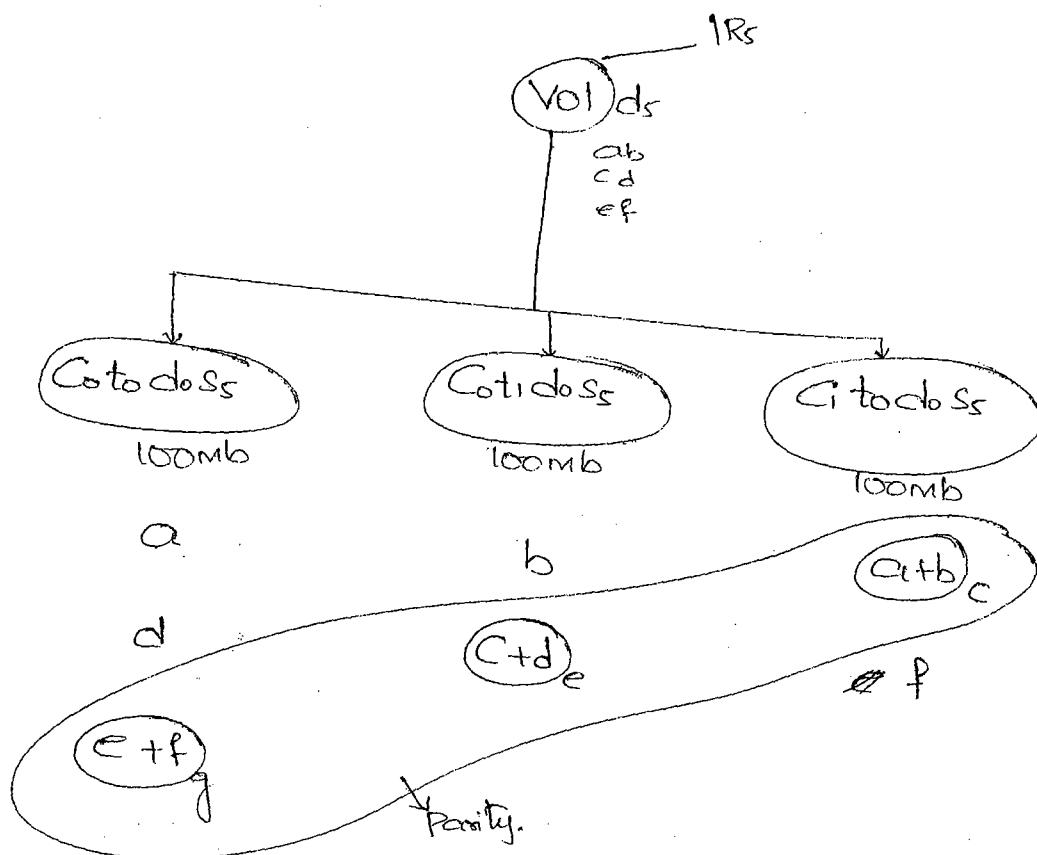
## RAID-1 := (Mirroring)



- High availability & low Performance.
- If any one of the Sub volume fails still we can Perform I/o operations.
- By default RAID Technique b/w Sub volumes is Mirroring.
- Slices should have same size.
- At max we can create 32 redundant copies.
- By default 2 way mirroring.

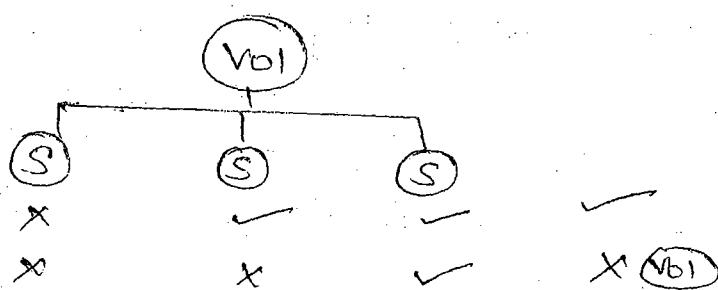
## \* RAID-5 (data with Parity)

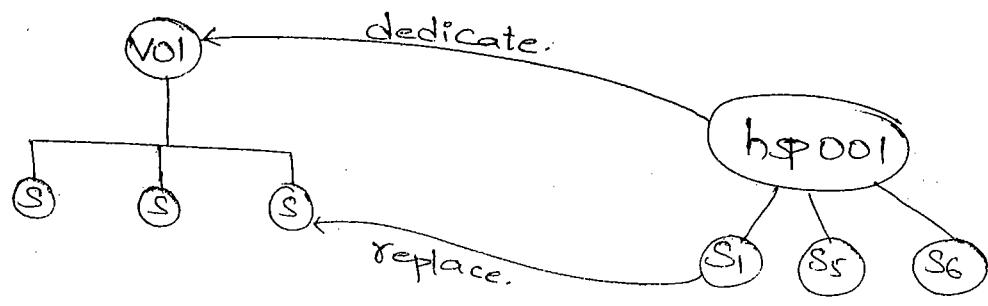
- Min. we have to initialize 3 volumes.



- In RAID 5 ~~we can~~ if 1 disk is Corrupted we can Perform I/O operations.
- If 2 disks are Corrupted Entire Volume is Corrupted.

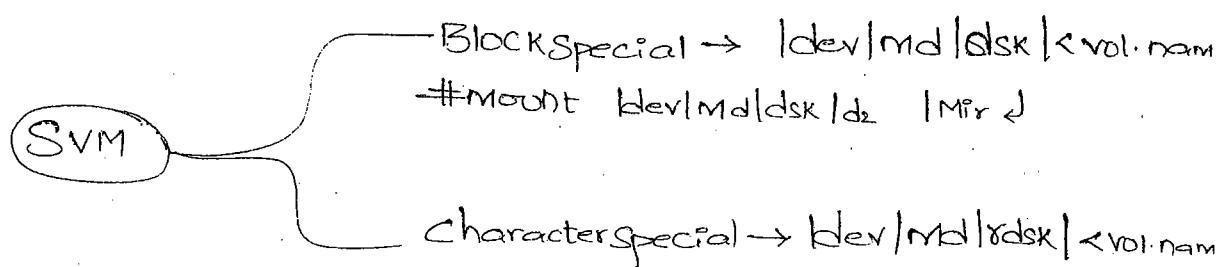
## Hot Spare Pool with RAID 5 Volume :-





## Device Naming Convention for SVM :-

SVM Provides two types of devices.



31/10/01

## SVM DEMO :-

- Take same H/w Config HD i.e., Same vendor & Same Cylinders.
- Attach a disk array to System.

# devfsadm -c disks ↴

# Format ↴

Assign Partitioning info.

Specify disk : 2

Format > P

Partition > P

Assign the Partitions 0-7 Slices 100Mb.

give 10mb for 7<sup>th</sup> slice. (for state db).

Partition > l ↴

y ↴

Copy the Partitioning info. to Other disks.

to see Partition info.

# Prvttoc [dev] rdsck | c1 t1 d0 s2 > file ↴

# fmthard ↴ : to Copy Partitioning info. to 2 (or) More disks.

# fmthard -s [file] [dev] rdsck | c1 t2 d0 s2 ↴

# fmthard -s [file] [dev] rdsck | c1 t3 d0 s2 ↴

# fmthard -s [file] [dev] rdsck | c1 t4 d0 s2 ↴

## Creating State database:

# metadb ↴ is a command used to Create (or) delete State database.

```
# metadb -a -f /dev/rdsk/c1t1d0s7 ↴  
      ↓      ↓  
      adding   forcefully
```

#metadb ↴ : also display Statedb info. (Created (or) not)

```
# cd /etc/lvm ↴
```

# [cat mddb.cf] ↴ is a file responsible to maintain Currently available Statedb info.

## Adding the Statedb :

```
# metadb -a -f /dev/rdsk/c1t1d0s7 ↴  
      ↴
```

## How to delete a State db.

```
#Metadb -d /dev/rdsk/c1t1d0s7 ↴
```

## Now Let us try to Create Logical Volumes:

### Creating RAID-0 Concatenation Volume:

System

```
# Metainit <volname> <no. of Components> <Component1> <SliceName>
```

# Metainit do 2 1 citidoSo 1 citedoSo ↴  
↓ ↓  
vol.name noofslices.

# Metastat -P ↴ : Used to list the available volumes  
Syntax info.

# Metastat do ↴ : Used to display Volume Statistics.  
\* When we execute metastat -P cmd. internally it reads  
[etc/lvm] md.cf file

# newfs /dev/md/rdsk/do ↴

# mkdir /Concat ↴

# mount /dev/md/dsk/do /Concat ↴

# df -h ↴

to Increase the Size of logical Volume (Concatenation)  
online:

# Metattach <sup>existing</sup> <vol.name> <newComponent>

# metattach do cit2.dos1 ↴

# Metastat do

\* When we add the ~~new~~ Component to an existing Volume. It updates only on Volume level. It won't update filesystem level.

\* In order to update the filesystem level run growfs cmd.

```
# growfs -M /Concat /dev/md/rdsk/do
```

## How to Remove a Concatenation Volume

`#metadclear` ↳ used to remove logical volumes.

`#metadclear do` ↳  
↓  
volname

`#metastat -P` ↳

## Creating Striped Volume in RAID-0:

`#metainit <Vol.name> <no. of Stripes> <no. of Components>`

<Component 1> ...

`#metainit di 1 2 GtidoSo Gt2doSo` ↳  
↑  
no.ofstripes  
↓  
slice default

`#metastat di` ↳

To increase the Interlace value at the time of creation, only Possil

`# metainit di 1 2 GtidoSo Gt2doSo -i 64K`

\* newfs /dev/ld1/rdsk/ld1

\* mkdir lstripe

\* mount /dev/ld1/rdsk/ld1 lstripe

\* df -h

# Cd 1stripe.

# touch \$1 \$2 \$3

Now remove a slice that is in striping.  
then try to access (or) create files we cannot.

### Deleting the Striped Volume

# umount 1stripe

# Metaclear d1

Note: Try to create 2 Concat  
Volumes & initialize a  
Volumes into Parent Vol.

### Creating RAID-1 Mirrored Volume:

Create 2 Concat Vol.

# Metainit d<sub>1</sub> 1 1 C1t1d0s0

# Metainit d<sub>1</sub> 1 1 C1t2d0s0

### Initializing Stripe Volume under Parent Volume.

# Metainit d1o -m d0

↓  
Submirror

# Metastat d1o

next

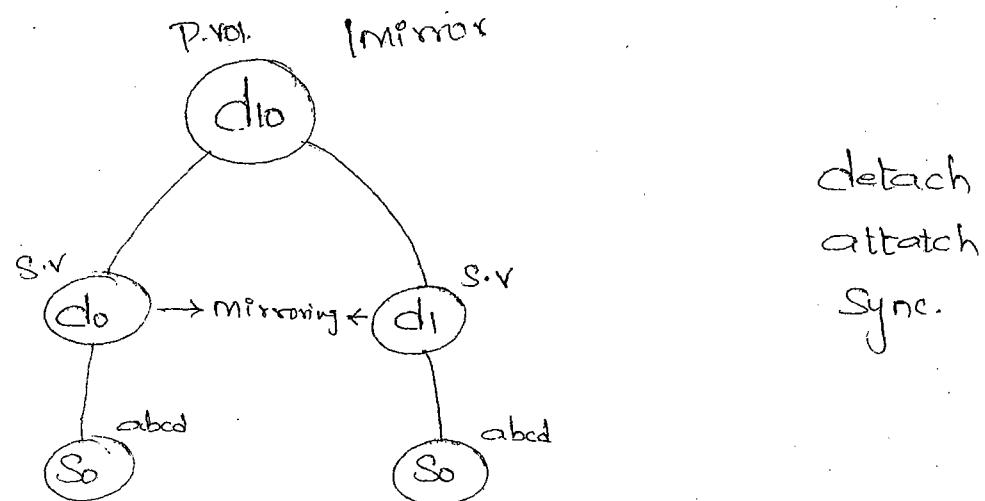
# Metattach d1o d1

# Metastat -P

```

# Newfs /dev/md/rdsk/dio
# mkdir /mirror
# Mount /dev/md/rdsk/dio /mirror
# df -h

```



```

# cd /mirror
# touch a b c d
# Metastat -p

```

detaching the Subvol. from existing Vol.

# Metadetach dio do

# Metastat -p

# mkdir /testmirror

# Mount /dev/md/rdsk/do /testmirror

# df -h

# cd /testmirror

# ls

# cd /mirror

# ls

```
# cd /testmirror
```

```
# touch 1 2 3 4 ↵
```

```
# cd /mirror
```

```
# touch new1 new2 new3 ↵
```

```
# umount /testmirror ↵
```

Reattaching Subvolume to existing Parent vol.

```
# metattach dio do
```

```
# metastat | grep -i sync
```

```
# cd /mirror
```

```
# ls
```

~~# sync~~ data will sync from Existing Vol. to newly added Vol.

How to remove mirrored Volume ?

```
# umount /mirror
```

```
# metaclear dio ↵
```

```
# metaclear -r dio ↵ : Will remove Submirrored Copies also.
```

Creating RAID → VOLUME

Metainit ds -r C1t1d0s0 C1t2d0s0 C1t3d0s1  
↓  
to create raid5 vol.

Syntax:

# Metainit <Vol.names> <Options> <Comp..1> <Comp..2> <Comp..3>

# Devofs /dev/1md1/8dsk/1ds ↳

# mkdir /raids ↳

# mount /dev/1md1/dsk/1ds /raids ↳

# df -h | grep -i /raids ↳

In Raid5 vol. We Cannot get back the data if 2 disks are Corrupted Even though we have More than 3 disks.

For that we go with Hotspare Pool.

# cd /raids

# ls

# touch ff kk ll pp uu yy

try to corrupt any one slice Manually.

# format

{  
make the slice 0.

```
# cd /raid5
```

```
# cp . /etc/
```

```
# metastat ds <
```

```
# ls <
```

## How to replace a failed Component:

```
# metareplace ds citido50 citido51
```

```
# Metastat | grep -i sync.
```

```
# Metastat ds
```

## How to Create HotSpare Pool:-

```
# Metainit <hspool.name> Com1 Com2 ...
```

```
# Metainit hspool citido53
```

```
# Metastat hspool
```

## How to add a New Component to an Existing hotSpare Pool.

```
# metahs -a hspool citido53
```

How to delete a Component in existing hot Spare Pool.

```
#Metahs -d hspool C1t2d0s3 ↵
```

Dedicating a Hotspare Pool to existing RAIDS Volume:

```
# Metaparam -h hspool ok
```

↓  
Specify  
hotspare

```
# Metaparam -h none ds ↵
```

to remove hotspare  
Pool

```
# Metastat ds
```

Goto format utility & Corrupt any one of the Slice Partitions in raids

```
#Metastat ds ↵
```

```
# Cd /raids
```

```
# CP . /usr/* ↵
```

```
#Metastat ds ↵
```

How to Replace a hotspare Pool with new Componer

-e <sup>enable</sup> ds C1t0d0s1

```
# metaReplace oldpool C1t0d0s1
```

```
# Metastat ds ↵
```

Metareplace ds C1t0d0s0 C1t0d0s1

<Corrupt comp> <new comp> 178

Create s1 in format ag  
Eatsign on

## Soft Partition :

# format ↳

Create a Partition with 1GB Space.

Label it

don't execute newfs Command.

# metainit <Softpartition Name> <Options> <Source of Sfppt·slice name>

<Size> ↳

# metainit d99 -P /dev/rdsk/c1t2d0s0 500mb

# newfs /dev/md/rdsk/d99

# mkdir /sfp1

# Mount /dev/md/dsk/d99 /sfp1

# df -h | grep -i /sfp1

How to increase Softpartition Size.

# Metattach d99 1g

# growfs -M /sfp1 /dev/md/rdsk/d99

# df -h | grep -i /sfp1



## Lab: ABP (Alternate Boot Path)

# Format ↴ : to know how many hd-disks

# eeprom boot device ↴ : to know bootable disk info.

# dd if = /dev/rdsk/c0t0d0s2 of = /dev/rdsk/C0t1d0s2 ↴

bs=2048 ↴

'Inout Page'

Goto OK boot Prompt.

Change the Path

OK > Print env ↴

bootable disk info.

OK > quit

OK > devalias ↴

Check alias name of Hd2 & Hd1

OK > Set env boot-device disk1. net

OK > quit ↴

OK > boot disk1

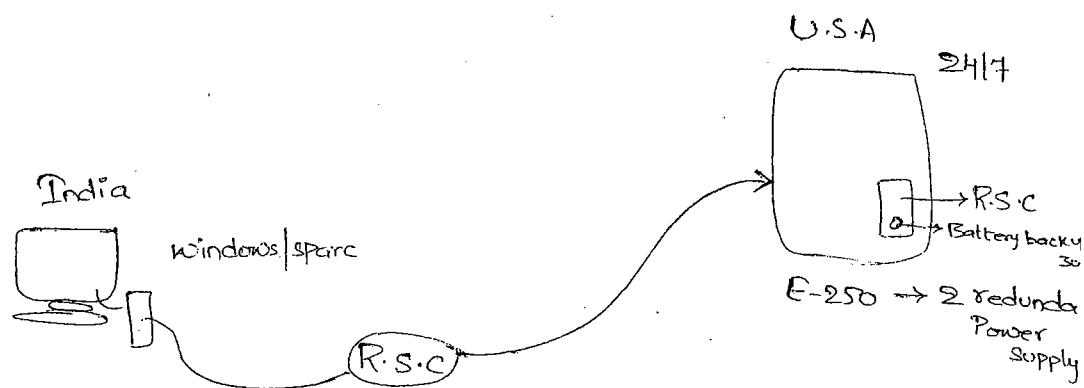
2/11/01

## K.S.C (Remote Server Control)

- Complete / Max. Administration can done on R.S.C only.

RSC Controller       $\xrightarrow{\text{Advanced version}}$  "Alomes" (Advanced light out Machines)

↓  
E-250  
V440  
V480



RSC > Poweron ↴  
> Poweroff ↴

- Using Rsc Controller we can Completely Monitor the Remote System.

- Rsc is a Piece of Hardware which is Used to Provide Remote System Administration.
- Rsc is Server Management tool that allows you to Monitor & Controls the Server over Modem lines & Over (or ethernet) the network.

- The Package for Rsc is **SUNWRS**
  - Manually we need to download from Sunsolve.com

- Once you install the Package it automatically updates

```
# cd /usr/Platform | 'uname -m' | ↵  
ls ↵  
RSC
```

- SUN E-250 Rsc can have two different types of Communication Ports.
  1. Through Modem lines
  2. Ethernet Network.

- Rsc can support two types of Configurations
  - \* GUI
  - \* CLI (`rscadm`)

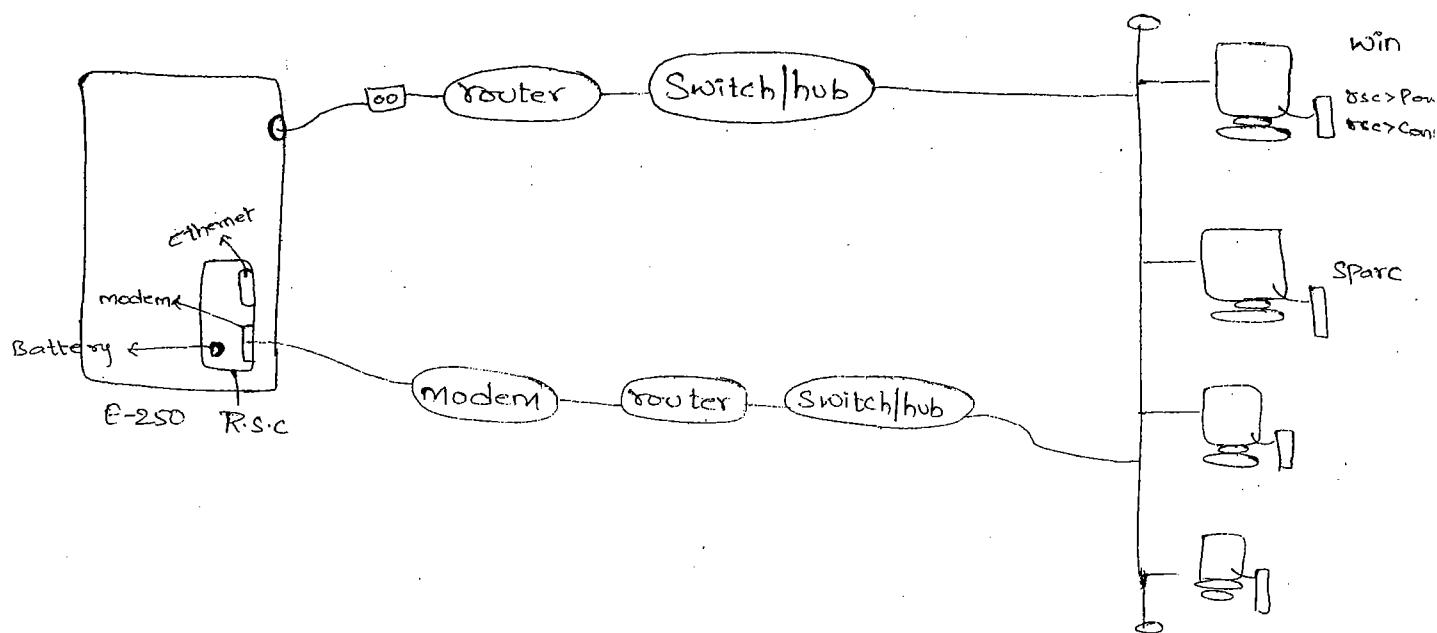
If we want to Access GUI then we need to install `rscclient` S/o.

- Rsc also have battery backup approximately 30 min. in case of Power failure.

- We can access Rsc from Remote Systems it may be Workstation, Windows Platform etc.

- Rsc can have its own Nic Card, battery backup, Processor, I/O chip & two Communication Ports.

- After installing Rsc SW on Server, we can Configure, Manage, Reboot, diagnostic from a Remote Server Console.



E-250 Servers doesn't have any internal Modems. We need to add externally.

- Rsc Controller Card can support at Max. 4 Authenticated User

## RSC Features :-

- Remote System Monitoring & Error Reporting. Including output from O.B.P that means we can observe complete remote system booting procedure.
- Remote Server Reboot, Poweroff, Poweron on demand.
- Ability to Monitor fan Sensors and temperature of CPU's, disks etc.
- Ability to run diagnostic test from Remote Console.

- RSC Battery backup allows the user after the complete failure
  - RSC Configuration defines & enables alert mechanisms.
  - Alerts are provide remote notification of system problems and it can be send to pagers (or) E-mail addresses.
  - RSC sends alert messages to authorized persons whenever the crashes are occurred.
    - E.g. If the <sup>remote system</sup> Temperature is low/high
    - \* Server reboots
    - \* If a Redundant Power Supply fails.
    - \* If the RSC Card battery backup is low.
  - We can access the RSC through Modem lines also with the help of **TAP** Protocol.
  - TAP Stands for telocator Alpha Numeric Protocol.
    - RSC-config**  
**rscadm**
  - Even you can monitor the remote system LED's also.
- >Environment** : used to display LED (light emitting diode)
- >Set locator** (only on VME server) : used to turn off/on LED's
- >>Show locator** : used to display whether the LED's on/off.

SMTP } Protocols used to send Mails & Alerts to the Users.  
SNMP }

- Continuously the Password of authorized users will be changing.

## How to Create Rsc Authenticated User:

# ./rscadm usershow ↴ : Shows all Rsc authenticated users.

# ./rscadm useradd <username> ↴

# ./rscadm userpasswd <password>

# ./rscadm userpasswd <username> ↴

New:

Retype:

# ./rscadm userperm <username> <perm> ↴

### Permission Types:

C - Connect the console

U - Setup other User Opt

Q - Setup Config. Variables.

R - Reboot, Poweroff, Poweron.

Note: When we are accessing the R.s.c Controller before that we have to place System in Nonsecure mode.

(In Physical)

## Questions

+ `fstype` → UFS

in Kilobytes - `df -k`

- Mount

`[etc/mnttab]` -

~~Do~~

How might you determine to which mount point directory  
the unmounted `/dev/dsk/c0d0s6` filesystem was  
last mounted.

`ls -ald /dev/dsk/c0d0s6`

`[who -a]`

`whodo` : Used to display currently logged users &  
their process info.

`[last]`

`boot -a` : Interactive booting.

[Who -r]: Shows Current runlevel

— If you want to start any application automatically  
Write a script under /rc2.d

Killing Scripts (K)

Starting Scripts (S)

3/11/09

## VERITAS

(Vxfs), (Vxvm)

- Veritas is an 3<sup>rd</sup> Party application sw which is Provided by the Semantic Organization.
- All the Packages in Veritas Should Start with "VRTSxxxx"

### The Packages

- VRTS vxvm — Volume Manager
- VRTS vxfs — filesystem.

### Drawbacks in Traditional file System :-

- online administration is Possible only with some limitations.
- one Single filesystem Cannot Span across Multiple Systems.
- "Relayouts" are not Possible. ( $\text{Raid} \xrightarrow{x} \text{Raid} s$ )
- limited no; of "inodes".
- limitations of 0-7 Slices.
- we can't Implement Raid levels in TFS. only with SVM we can.

## Features of VxVM , Vxfs :

- online administration is Possible.
- one single filesystem can span across multiple disks.
- online re-layout's are possible
- online backups & snapshots.
- DRL , Raid-5 log. (increased to system performance).  
(Dirty Region Log)  
only on Raid-5
- we can implement quotas.
- have extended based filesystem.
- Exporting & Importing is possible.

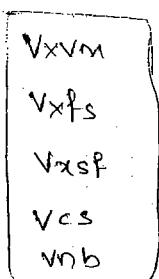
increase file size  
(Vxresize)

decrease file size.

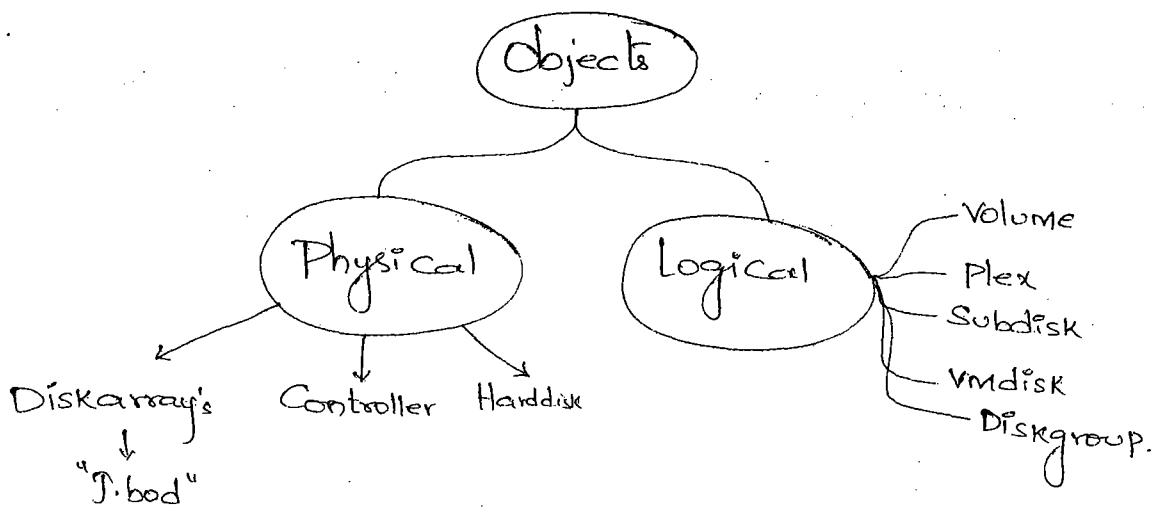
We can configure Veritas Volume Manager in 2 ways

Graphical User Interface → # `Vreal`

Command line Interface → # `Vxmake`  
# `Vxassist`



## Veritas Objects :-



## Physical Objects :-

### Diskarrays:

- We can call as T.bod (or) bunch of disks.
- Collection of 2 (or) More disks

### Hard disk:

- used to store data | Create files.

### Controller:

- Interface b/w harddisk & CPU Unit
- Responsible for maintain 2 (or) More disks.

## Logical Objects:

Subdisk: - Is a logical Collection of Contiguous disk blocks.

- Subdisks Cannot Span across Multiple disks.
- generally Subdisks overlapping is not Possible

## Plex.

- Subdisk default naming convention is "diskname#-#"
- The default Raid technique b/w subdisks is Concatenation.
- We can combine different subdisks from different physical disks & we can create a logical volume.

Plex: - is a combination of 2 (or) more subdisks.

- at Max. we can create 32 plexes in a one single volume.
- The default Raid technique b/w two (or) more plexes is "Mirroring".

Plex naming convention is "Volumename#-#"

- ultimately plex can define what type of RAID technique we want to implement in a subdisk.

Volume:

- It's a combination of two (or) more plexes.
- It can support max. 32 plexes in a one single volume
- A Volume can span across multiple disks.

The default naming convention for Volume

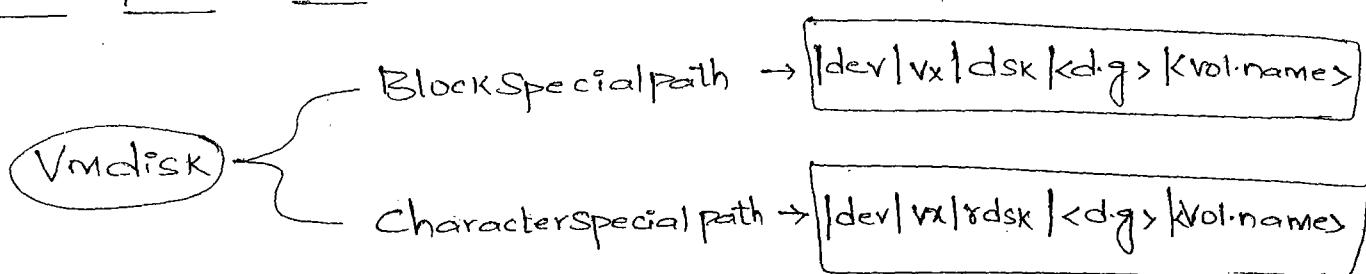
"Volumename #"

## VMDisk: (Virtual Mgt disk)

- When we initialize a physical disk under the control of Veritas Volume Manager the disk can be called as VMDisk.

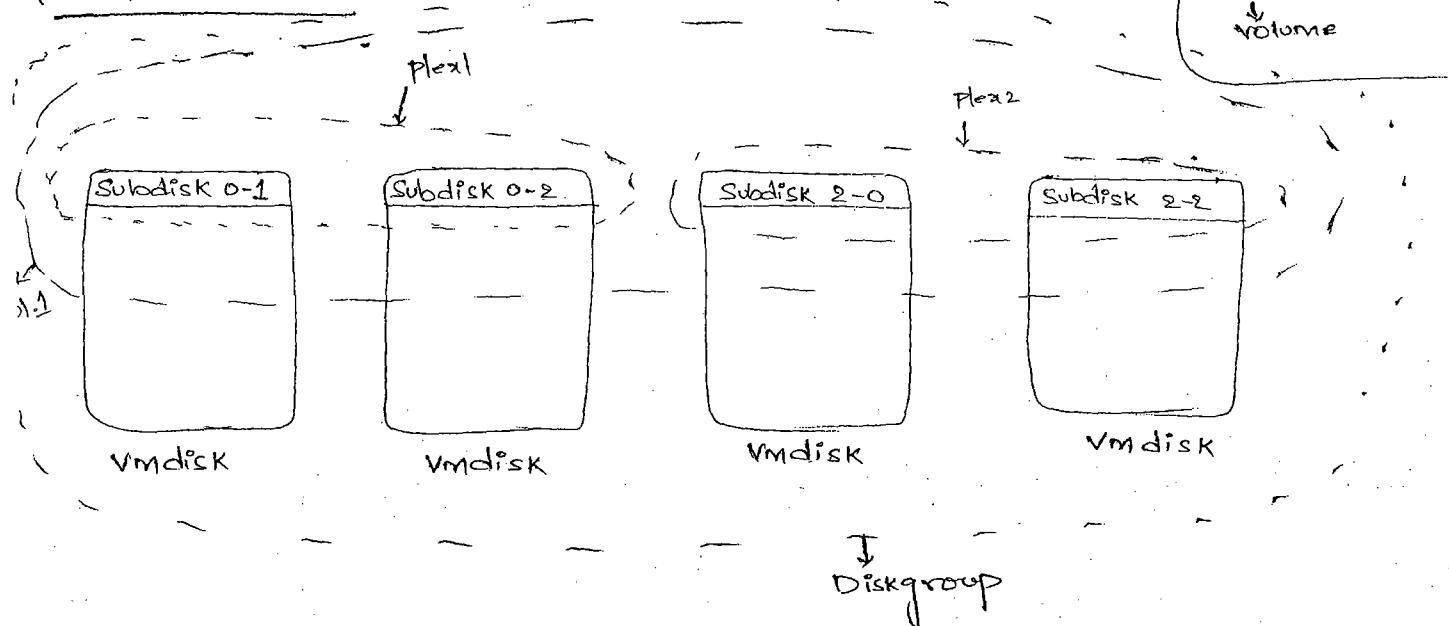
Diskgroup: - It's a combination of 2 or more VMDisks

Device path for VMDisk:

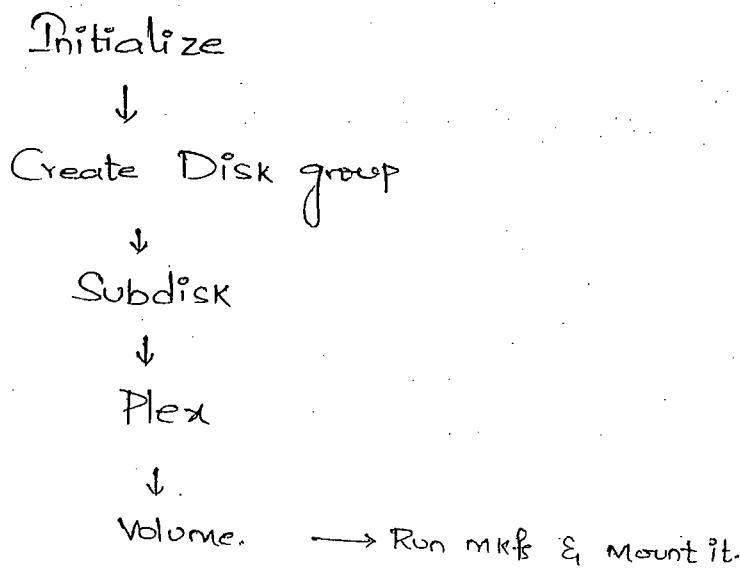


#mkfs is used to build Veritas file system on raw path.

Architecture:



## Hierarchy:



## Initializing the disks:-

- There are 3 ways to initialize disk under Veritas Vol. mngt Control.

1. Sliced
2. Simple
3. Cds

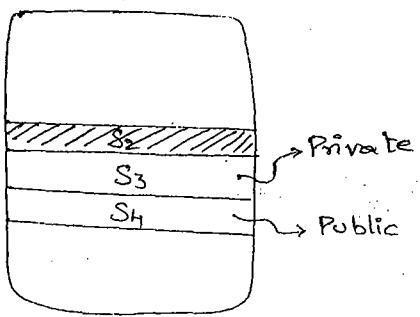
1. Sliced : - generally we have to initialize the disk for traditional file system mirroring.

- When we initialize the disks under Sliced format automatically disk can be divided into two regions.

(i) Public Region (ii) Private Region

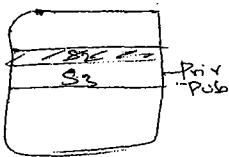
↓  
~~Physical Metadb~~  
- generally used to store the data.

↓  
- Similar to metadb in SVM.  
- Responsible for maintain Vol's & config. info.



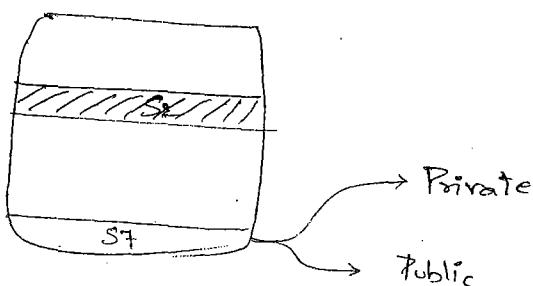
## 2. Simple:

- When we initialize disk under Simple format the Public & Private regions are created under one single slice.
- It Won't Support heterogeneous platforms.



## 5. Cds: (Cross platform data storage)

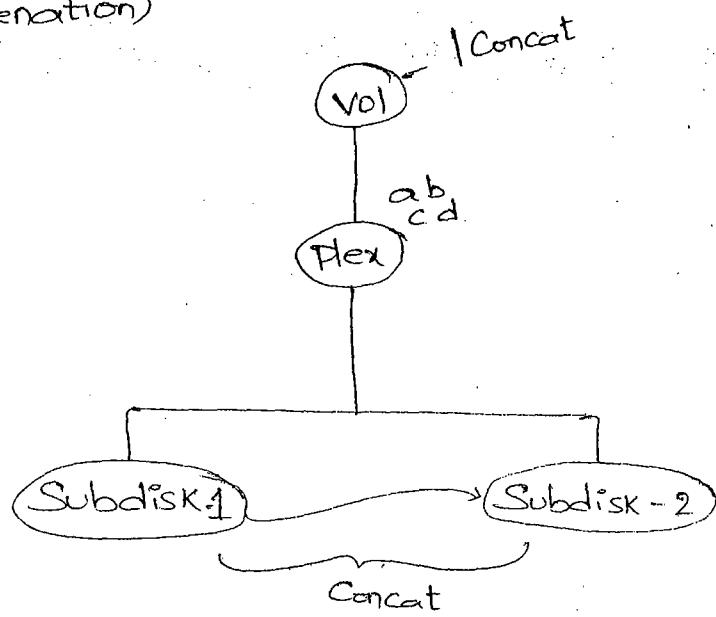
- It can support heterogeneous platforms.
- When we initialize disk under Cds format. The Private & Public regions are updated in a one single slice i.e., S7 slice.



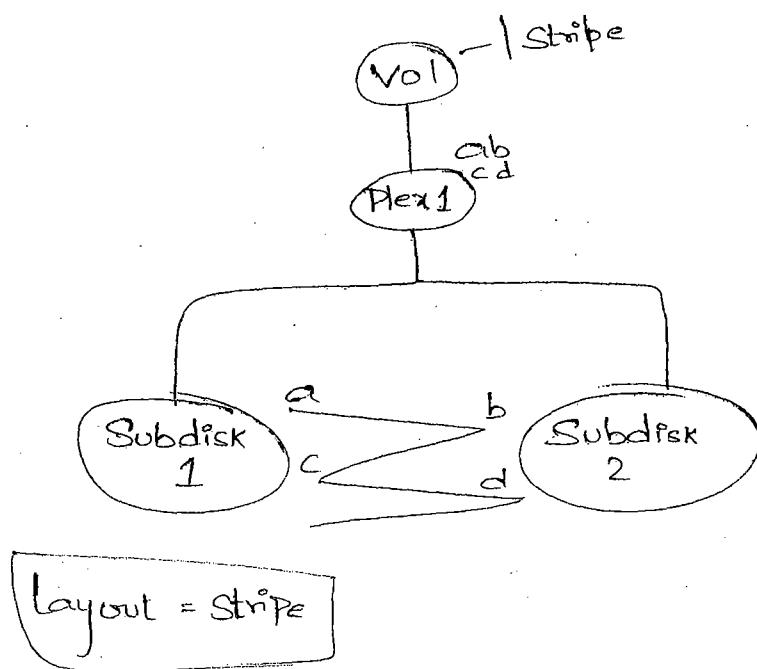
- It's a default format for all Vm disks.

## KAID Levels:

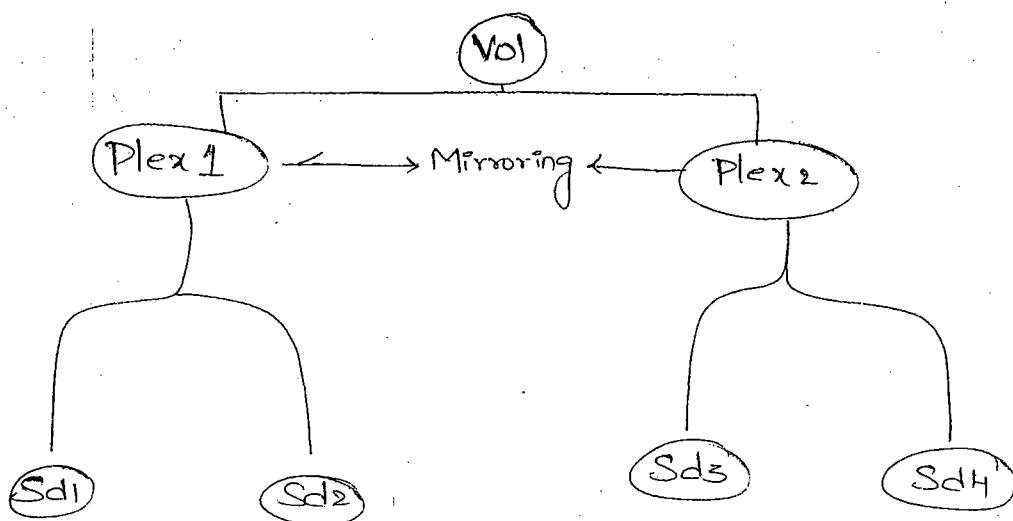
### Raid - 0 :- (Concatenation)



### Raid - 0: (Striped)

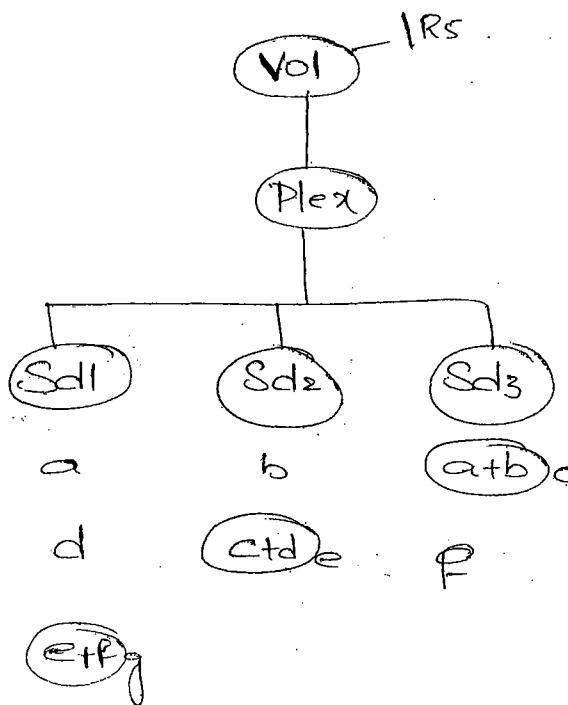


## Raid - 1 : (Mirrored Volume)



Raid 1 to  
0+1

## Raid - 5 :



## Deamons for vxvm :

- Vxconfigd: - It's a main Configurational deamon. Responsible for running all the Volumes.
- If the Vxconfigd is not running, Manually we have to Startdeamon.

# Vxconfigd ↴

- If Vxconfigd is in accessible we can't interact with any Volumes.

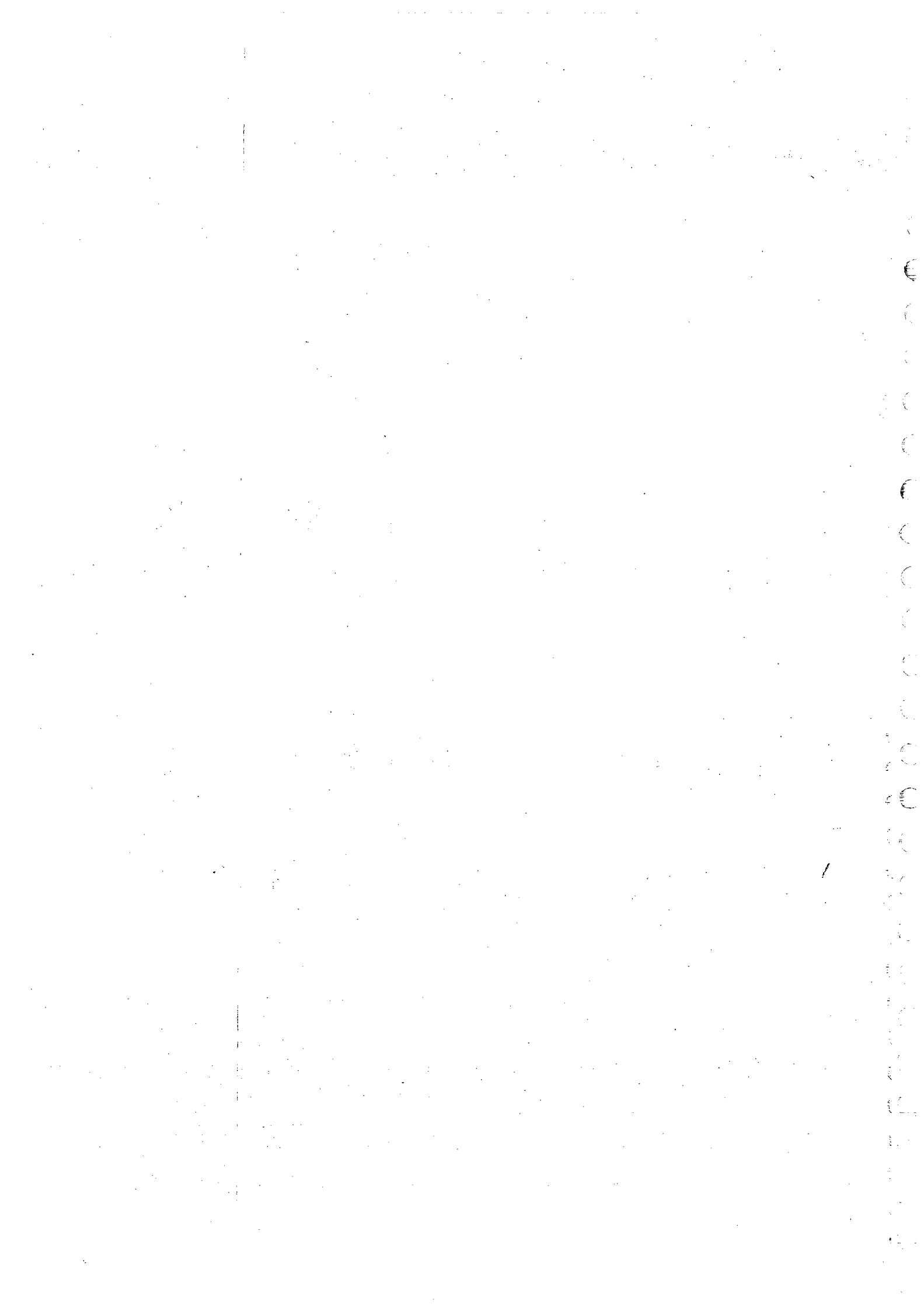
# Vxdctl mode ↴

# Vxdctl enable ↴

- Vxiod: - Responsible for Mountain Veritas I/O Operations.

- Vxnotifyd: - Responsible for generate the Notice msgs. When we Initialize (or) Uninitialize disks.

- Vxrelod: - Responsible for replace the Corrupted Components.



## Lab Project: Root Mirroring

|etc|Vfstab

|etc|system - Kernel Parameters

# cp |etc|system |etc|System.bkp ↴

# cp |etc|Vfstab |etc|Vfstab.bkp ↴

# Metainit do || Cotodos ↴

# Metainit -f do || Cotodos ↴

# Metainit do -M do ↴

# df -h ↴ ~~metainit do || Cotodos~~

# Metaroot do ↴

# df -h ↴

# init 6 ↴

# Metainit di || Cotodos ↴

# Metastat -c

# Metattach db di ↴

1/11/09

## DEMO: VxVM Software Installation

Min. Requirements:

- Create /opt file system with 3gb.
- Download the SW from CJs Server.
- Provide demo license key.
- change the date to 2000 year.

# Ping 200.200.0.22 ↴

# Showmount -e 200.200.0.22 ↴

# mkdir /test ↴

# Mount -f nfs 200.200.0.22:/opt /test ↴

# Cd /test ↴

# Cd SF4~1.1-S ↴

# Prod.

# Cp -r SF\_hacn1 /opt ↴

# Umount /test ↴

# Cd /opt ↴

# ls ↴

# Date 01/12/2000 ↴

# Uname -a ↴

```
# cd /opt <
```

```
# tar -xvf sf_hac~1
```

The Package for Veritas Volume Manager is VRTSVXVM

The Package for Veritas File System is VRTSVxfs

```
# ls <
```

```
# ./installer <
```

VERITAS Storage foundation 4.1

Enter a Selection : 1

Select a Product to Install : 3 (Veritas Volume Manager)

Enter the System names Separated by Spaces on which to

Install VxVM : wstsun4 <

---

Note: Simultaneously we can install VxVM SW on Multiple nodes (or) Systems.

If you want to install on multiple systems

we have to establish trusted relationship on multiple n

Press Enter to Continue : <

Infra Structure Packages

Press Enter to Continue : <

Select optional Packages to be installed on all Systems : 1

(Config values in GUI, Man pages and Providing documentation)

: ↵

## Installation Requirements

: ↵

### Installing

:

## Command locations for Veritas Volume:

- When we install Shw it automatically updates 2

locations. (i) /etc

(ii) /opt

/etc/vx/bin

/opt/vrts/bin

Actually to Create Veritas Volume we need to go to these locations & Create. In order to Create from Root level we need to Set the Path.

# vi /etc/profile ↵

Lastline

PATH=\$PATH:/etc/vx/bin:/opt/vrts/bin

Export PATH

MANPATH=\$MANPATH:/opt/vrts/man

Export MANPATH

:wq! ↵

#sh /etc/profile ↵

After completing 20 Steps Enter to Continue.

:)

Enter VxVM license key for wostsunh: irpg-Peq-Itd8-yqlv-8ph  
ycc

another license key : No

:)

Are you ready to Configure VxVM: y ↵

Do you want to setup the enclosure based naming Scheme  
n ↵

Note: If we say Yes

Press return to Continue: y ↵

Do you want to setup default disk group : n ↵

:)

|OPT|VRTS|Install|Logs

: Maintains installation logs.

# Pkginfo |grep VRTSVxVM ↵

Installing the Veritas filesystem (Vxfs):

# cd /opt

# ./installer

Enter Selection : 1

Select Product : 2

Enter the System names Separated by Spaces on which  
to install VxFS : hostsun4 ↴

Press [return] to Continue : ↴

Infrastructure Package:

: ↴

Additional License Key : n ↴

Optional Packages: l ↴

: ↴

[OPT\VRTS\install\log] : to see all VxFS logs.

[Shutdown -y -i6 -g0] : gracefull reboot.

#Pkginfo |grep VRTSVxfs ↴

#modinfo |grep -i Vx

: to know the Version of Veritas  
Volume Manager.

Set the Path Variable under /etc/profile:

#Vxd ↴

# devfsadm -c disks ↴

# Vxdisk scandisks ↴ : to detect the external disks.

# Format ↴

# Vxdisk list : Used to display currently available disks.

Initializing the disks :-

# Vxdisksetup -i C1todo

↓  
to install

# Vxdisk list ↴

# Format

:e

:P

:q

# How to Initialize a disk under Sliced format

# Vxdisksetup -i C1todo format=sliced

# Vxdisk list ↴

# Format ↴

Check.

# How to Initialize a disk Under Simple format

# Vxdisksetup -i C1todo format=Simple ↴

## How to uninitialize a disk:

```
# Vxdiskunsetup -c c1t2do Gt1do ↴
```

## Initialize the disk under cds format

```
# Vxdisksetup -i C1t1do ↴
```

```
# Vxdisksetup -i C1t2do ↴
```

## Creating a disk group:

```
# Vxdg init oradg do=c1t1do ↴  
↓  
<disknames>  
group
```

```
# Vxdisk list ↴
```

## How to remove a disk group:

```
# Vxdg destroy oradg ↴
```

## How to add another disk to an existing diskgroup

```
# Vxdg -g oradg adddisk d1=Gt1do ↴ d2=C1t2do d3=C1t3do
```

```
# Vxdisk list ↴
```

## How to remove a single disk from an existing disk group

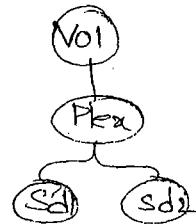
```
# Vxdg -g oradg rmdisk d3
```

## Creating Raid-0 Concatenation Volume.

Subdisk → Plex → Volume → Start Volume → mkfs → Mount.

### VxMake:

#### Creating a Subdisk



```
# Vxmake -g oradg sd sd1 do,0,100M ↴
```

for next 100m, 700m

```
# Vxprint -St ↴ : to know Subdisk Created (or) not
```

#### Creating a Plex

```
# Vxmake -g oradg Plex plex1 sd=sd1 ↴
```

```
# Vxprint -Pst ↴
```

table of cont.  
Plex subdisk

```
Vxplex -g hasiom dis plex ↴
```

To remove Plex first we  
need to dissociate

```
Vxedit -g hasiom -rf rm plex ↴
```

#### Creating a Volume

```
# Vxmake -g oradg Vol contrl Plex=plex1 ↴
```

```
# Vxprint -Vpst ↴
```

Adding Plex to  
vol.

```
Vxplex -g s...  
att mirv ↴
```

#### How to Start the Volume

```
# Vxvol -g oradg Start contrl ↴
```

```
# Vxprint -hvps ↴
```

By default the User type for Vxmake is gen

Build the filesystem.

```
# mkfs -F vxfs |dev|vx|rdsk|oradv|Convol|
```

```
# mkdir |Concat
```

```
# Mount -f vxfs |dev|vx|dsks|oradv|convol|Concat
```

How to Remove Concatenation Volume:

```
# Vxedit -g oradv -rf &rm Convol
```

recursively  
&  
forcefully

```
# Vxprint -hvpst
```

Creation of Raid-0 Striped Volume

```
# Vxmake -g oradv sd1 sd1 do,0,100M
```

```
# Vxmake -g oradv sd1 sd2 do,0,100M
```

Plex:

```
# Vxmake -g oradv plex plex1 sd=sd1,sd2 layout=stripe
```

stripwidth = 64  
Interleave value

Volume:

```
# Vxmake -g oradv vol Strvol Plex=plex1
```

```
# Vxvol -g oradv Start Strvol
```

```
# Vxprint -hvpst
```

# mkfs -F vxfs /dev/vx/rdsk/oradg/stvol ↴

# mkdir /stripe ↴

# mount -F vxfs /dev/vx/dsk/oradg/stvol /stripe ↴

Removing the Striped volume:

# umount /stripe ↴

# vxedit -g oradg -f /m stvol ↴

Creating Raid-1 Mirrored volume:

# vmake -g oradg sd sd1 d0,0,100m

# vmake -g oradg sd sd2 d1,0,100m

Plexes:

# vxmake -g oradg plex plex1=sd1 ↴

# vxmake -g oradg plex plex2=sd2 ↴

Volume:

# vmake -g oradg vol mirvol plex=plex1,plex2

# vxvol -g oradg start mirvol ↴

# vxtask monitor ↴

to see what happening background. (6) Progr

Build a file system & Mount it ↴

## Removing the mirrored volume

# umount /mirror ↴

# vredit -g oradg -rf tm mirvol ↴

## Creating the Raid-5 Volume:



# vmake -g oradg sd sd1 d0,0,100M ↴

# vmake -g oradg sd sd2 d1,0,100M ↴

# vmake -g oradg sd sd3 d2,0,100M ↴

# vmake -g oradg plex plex1 Sd=sd1, sd2, sd3 layout=raid5  
st width=64 ↴

# vmake -g oradg -Uraid5 Vol rsrvol Plex=plex1 ↴  
↓  
usage type.

# rsrvol -g oradg start rsrvol ↴

# vxprint -hypst rsrvol ↴

Build the file system & Mount it ↴

## Removing the Raid-5 volume:

# vredit -g oradg -rf tm rsrvol ↴

## Uninstalling the VxVM:

#trapprint -hrpst ↴

remove the existing volumes

#vxdisk list ↴

#Vxdg destroy Oracle ↴

# vxdisk list ↴

# Vxdiskunsetup -c C1t0d0

# " -c C1t2d0

# " -c C1t3d0

#vi /etc/profile ↴

### Lastline

delete path variables.

:wq! ↴

# . /etc/profile ↴

# cd /opt ↴

# ./installer ↴

\*

Enter Selection : U ↴

Select Product : 3 ↴

:wstsun4 ↴

Remove packages: y ↴

:y ↴

:y ↴

Manually Uninstall VRTs also.

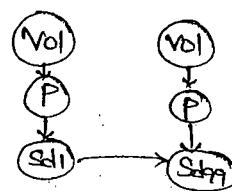
# Pkginfo | grep -i VRTs ↴

-- if Pkg exists delete them.

# Shutdown -y -i6 -g0 ↴

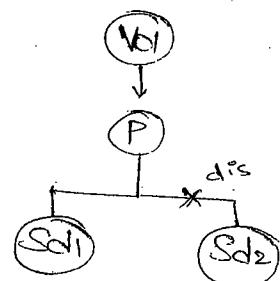
- It won't disturb I/O operations. Unlike the case of remove Subdisk.

remove:



~~#vxedit -g oradg -rf rm Sd1~~

# Vxsd :



dis:

~~#Vxsd -g oradg dis Sd2~~

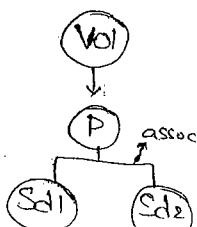
~~#Vxsd -g oradg -o force dis Sd2~~

to specify options

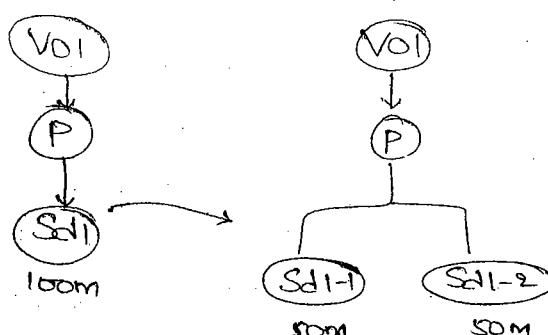
If Sd2 is last running subdisk then only we can remove this.

assoc:

~~# Vxsd -g oradg assoc <plexnames> <Subdisknames>~~



split:

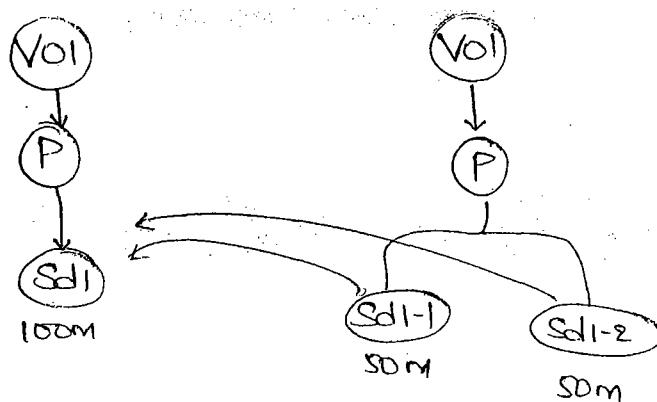


Split & Join doesn't Interrupt I/O operations.

~~# Vxsd -g oradg -s 50M split Sd1 Sd1-1 Sd1-2~~

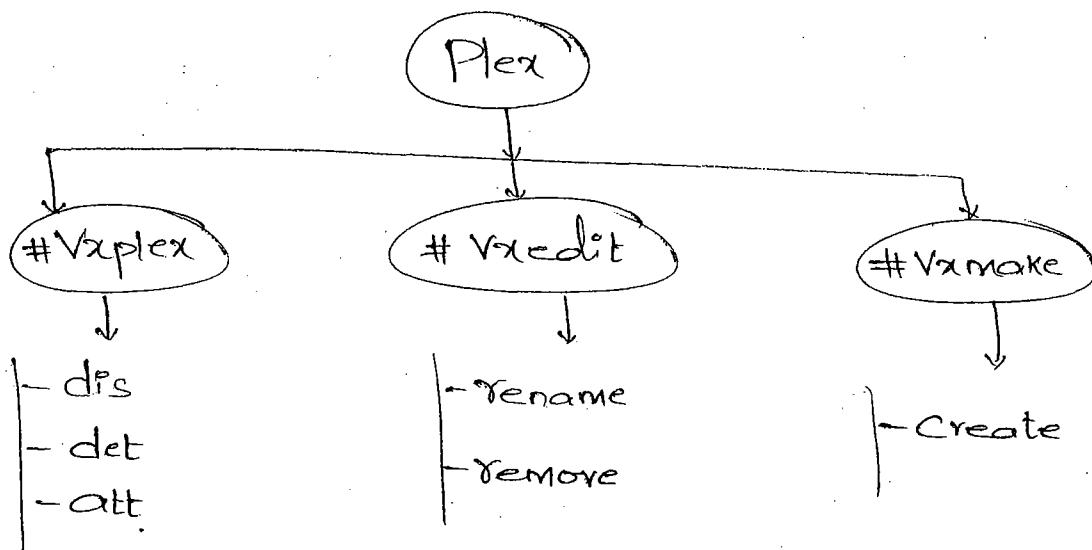
Spmpl size of first subdisk

Join



# Vxsd -g oradg join Sd1-1 Sd1-2 Sd1

Plex Level Operations: (Better to implement on Raid-1 only)

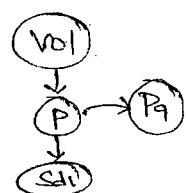


# Vxmake -g oradg Plex Flex1 Sd = Sd1 ↴

# Vxpoint -pt ↴

# Vredit:

- Rename



# Vredit -g oradg Rename Pg Pg ↴  
 ↓  
 Old name New name

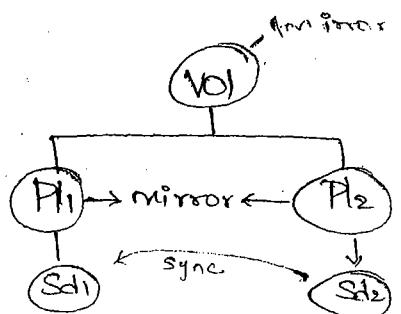
## - Removing

```
# Vxedit -q Oradv -rf dm <plexnames>
```

# Vxplex

dis

- Completely removing plex from the vol.



*det*

- det  
- Plea will be under control of vol. but there is no I/O opera

#Vaplex -g oradg det pli ↴

→ # Vplex → org dis PII ←

att: (How to attach plex to existing vol.)

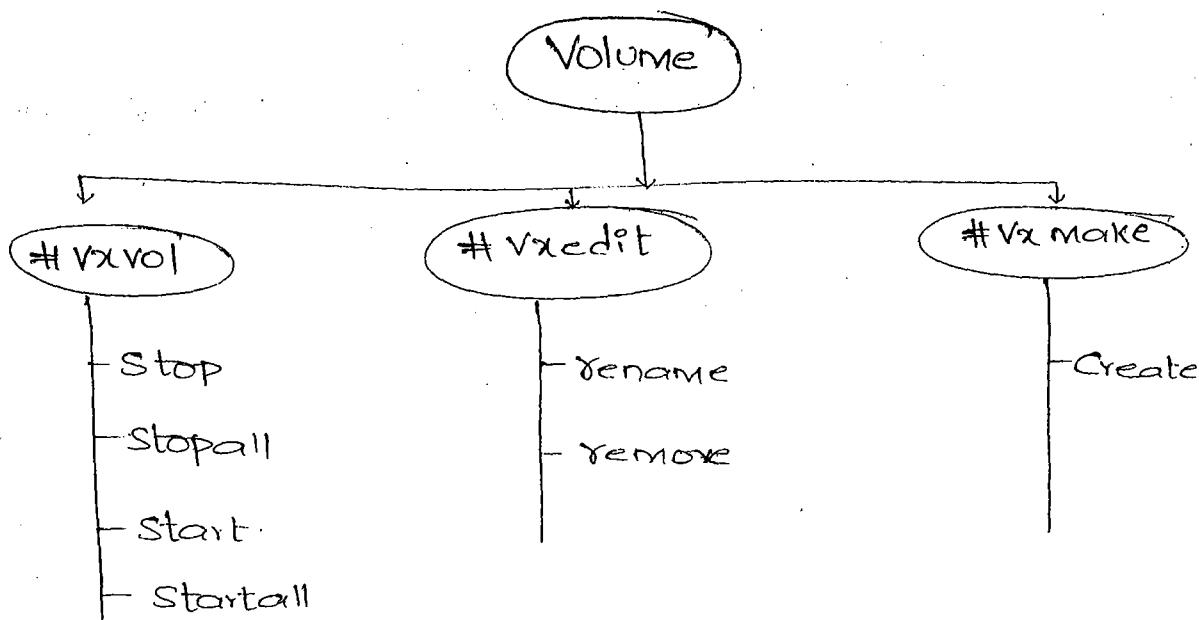
#Vplex -g oradg att <volnames> <plexnames> ↵

Before dissociating we need to make Plex offline.

`# VxMend` ] : Used to make a plex offline (or) online

# Valmend -g. Orady off pl. J  
" " " on pl. J

### 3. Volume Operations:



#### #Vxmake:

```
#Vxmake -g oradg Vol Voll Plex=Pl
```

#### #Vxedit:

rename

```
#Vxedit -g oradg Rename <oldname> <newname>
```

remove

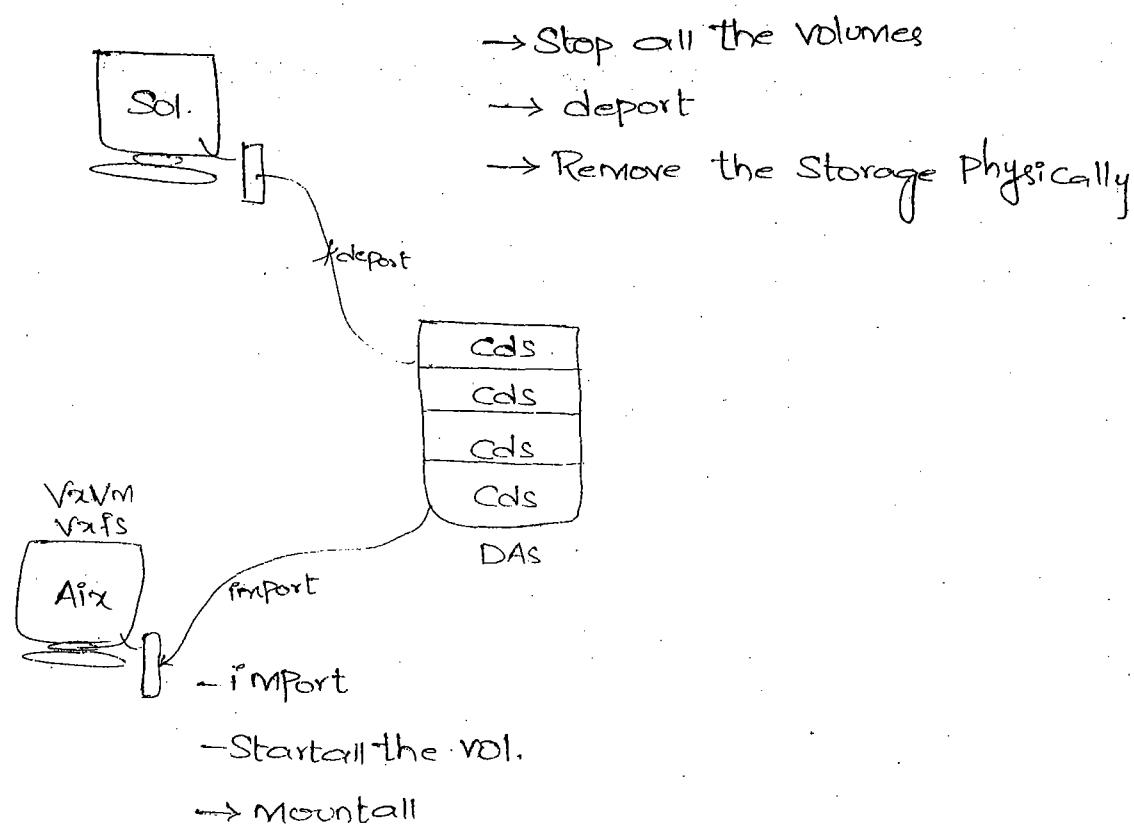
```
#Vxedit -g oradg -rf rm <volnames>
```

#### #Vxvol:

```
#Vxvol -g oradg Start Voll
```

```
#Vxvol -g oradg Stop Voll
```

## Deporting & Importing:



```
#vxvol -g oradg Stopall ↴
```

```
# Vxdg [ ] deport oradg ↴
```

```
#Vxdisk list ↴
```

→ Remove the DAS Storage Physically & attach to another System

```
#Vxdisk list ↴
```

```
# Vxdisk -o alldgs list ↴
```

: display all deported disk groups.

```
# Vxdg Import oradg ↴
```

```
#vxvol -g oradg Starialy ↴
```

Mount it.

#Vxassist: - is an intelligent cmd. Compared to #Vxmake.

- We can perform different things by using #Vxassist Command.

- \* Resize the Volumes
- \* Relayout's
- \* Relocation's
- \* Effectively Raid (0+1), (1+0)
- \* Online Snapshot

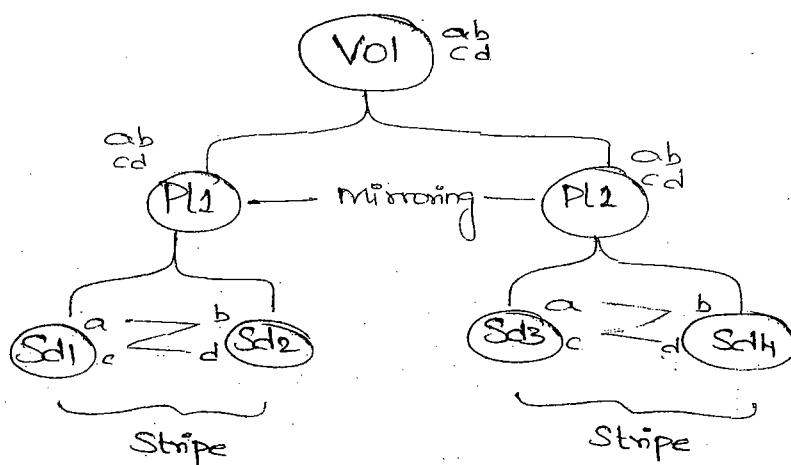
Vxmake

```
- Sd  
- Plex  
- Vol  
- Start  
- mkfs  
Mount
```

Vxassist

```
- Sd, Plex, Vol, Start  
- MK  
- Relocate fs  
Mount
```

Raid (0+1): (Mirror Stripe)



|   |   |   |   |
|---|---|---|---|
| x | ✓ | ✓ | ✓ |
| ✓ | x | ✓ | ✓ |
| x | ✓ | x | ✓ |

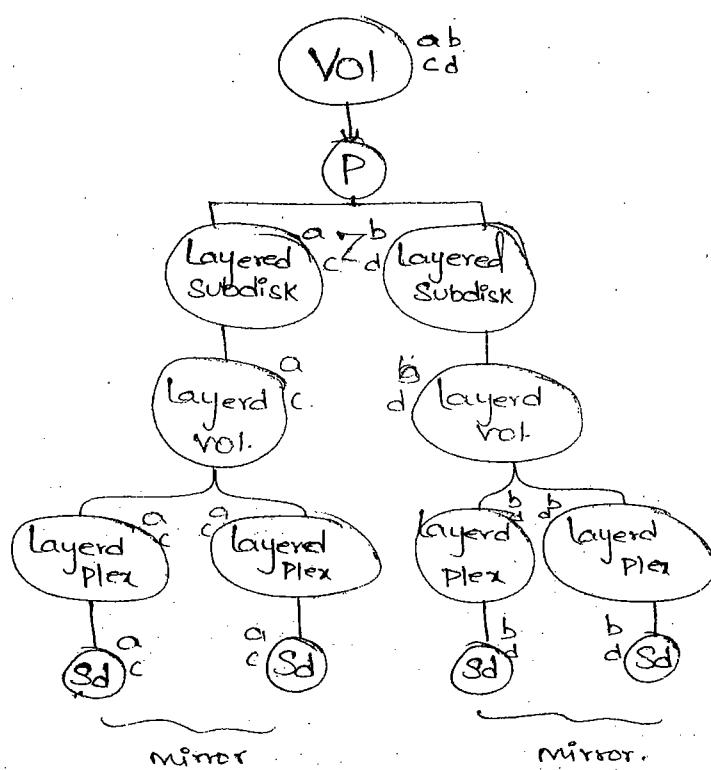
## Command for Creating Raid 0+1

```
#Vxassist -g oradg make <vol-name> 400M  
layout=mirror-stripe stripwidth=64 ↴
```

It automatically creates 2 plexes & subdisks with default naming conventions.

```
#Vxprint -A th ↴  
hvpst
```

Raid (1+0) :- Raid-10 (Striped mirror)  
"layered Concept"



```
#Vea & ↴  
to see hierarchy
```

## Command for Raid(1D)

```
# Vxassist -g Drdg Make Hoom <vol.names>  
Layout = stripe-mirror Swidth = 64
```

## Lab Project:

SSH: We can Secure the data in n/w.

Snoop: We can Capture the n/w info.

Sys1 IP: 2

# Snoop

WSTsun2 → WSTsun3 → login

Password:

Sys2 IP: 3

telnet 200.200.0.2

login

Passwd

## Configuring SSH:

# ssh -keygen -t rsa ↴  
↓ Creating algorithm.

Creates Public & Private Keys.

Yes ↴

# cd ssh

# pwd ↴

# ls ↴

# vi /etc/ssh/sshd\_config

goto 116 line

PermitEmptyPasswords no → yes

Permitrootlogin <sup>no ↗</sup> yes ↘

:wq!

# Svcadm restart ssh ↘

# Svcs ssh ↘

# Svcs -P ssh ↘

Daemon sshd  
Port no:

Sys1

Sys2

ssh 200.200.0.8

Yes

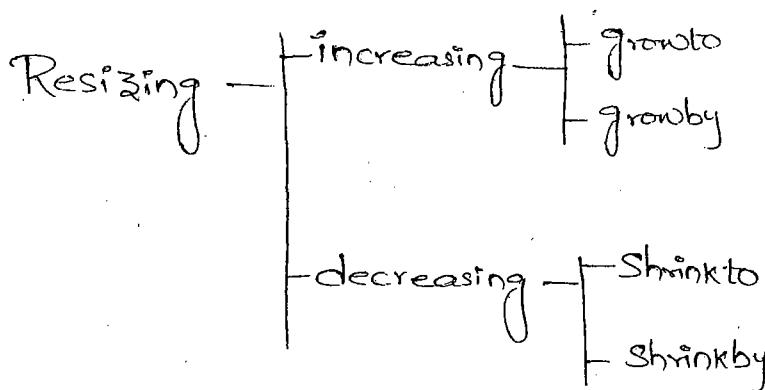
Password: ↗

6/11/01

## Resizing:

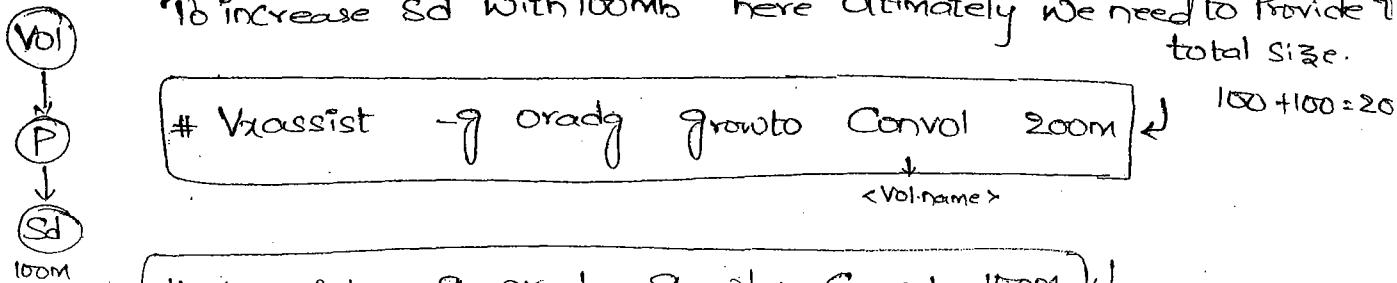
- Online we can increase & decrease filesystem size.

Note: When we are going for decreasing the filesystem size we can't give data guarantee.



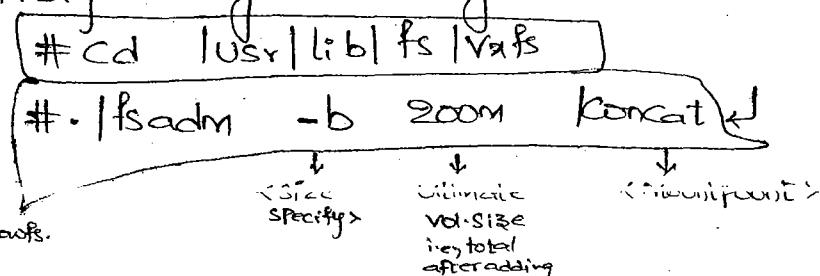
# VxResize : Used to increase (or) decrease the size.

To increase sd with 100mb here ultimately we need to provide total size.



# df -h

By Using growto & growby the size will be increased at Vol. level.  
For effectively change in filesystem level also we have to execute

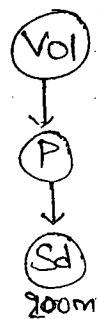


fsadm is like to growfs.

## How to decrease a fileSystem Size.

-P

```
# Vxassist -g Oracle Shrinkto Convol 100m ↴
```



to decrease the size ultimately we need to specify the size which is going to be there after decreasing.  
 $200 - 100 = 100m$ .

```
# Vxassist -g Oracle shrinkby Convol 50m ↴
```

↓  
which we need to decrease.

```
# ./usr/lib/ps/vxfs/psadm -b 100m | concat ↴
```

## VxResize :

Increase (or) decrease filesystem sizes recursively volume level as well as filesystem level.

```
# VxResize -g Oracle Convol +200m ↴
```

↓  
<Vol-names>

```
# VxResize -g Oracle Convol -200m ↴
```

```
fsck -f Vxfs /dev/vx/rdsk/oradv | testmir | testmir
```

↓  
vol.name

at mounting stage

If we get error, saying that Corrupted & needs Maintenance  
then execute

Error:  
= Cannot statrs. i.e., it is in Mounted Status.

Demo: #Vxassist :-

Creating Raid-0 Concatenation Vol.

```
# Vxassist -g oradv Make Convol 200M
```

```
# Vxprint -hypst Convol
```

Build f/s & Mount it.

```
# Vxredit -g oradv -sf &m Convol
```

Creating Raid-0 Striped Vol.

```
# Vxassist -g oradv Make Strvol 200M
```

layout = Stripe

Strwidth = 64

```
# Vxprint -hypst Strvol
```

Build f/s & Mount it.

## Creating Raid-1 Mirrored Vol.

# Vxassist -g oradg Make Mirvol zoom Layout=mirror ↴

Build the fls & Mount it.

## Creating Raid-5 Vol.

1st way

# Vxassist -g oradg Make vsvol zoom Layout=raids

Stwidth=64 ↴

Vxpoint -hvpt vsvol.

It will Create each Subdisk in all available disks.

-I don't want to generate the log & i want to specify with only 3 Subdisks.

# Vxassist -g oradg Make vsvol zoom Layout=raids,

No raids log Stwidth=64

Nc01=3 ↴

## Create Raid (0+1) Vol: (Mirrored Stripe)

# Vea& : in gui

Veritas enterprise administration.

# Vxassist -g oradg Make Mir-str zoom Layout=mirror-

Stripe Stwidth=64

# Verbase monitor

## ~~1110~~ SNAPSHOTS:

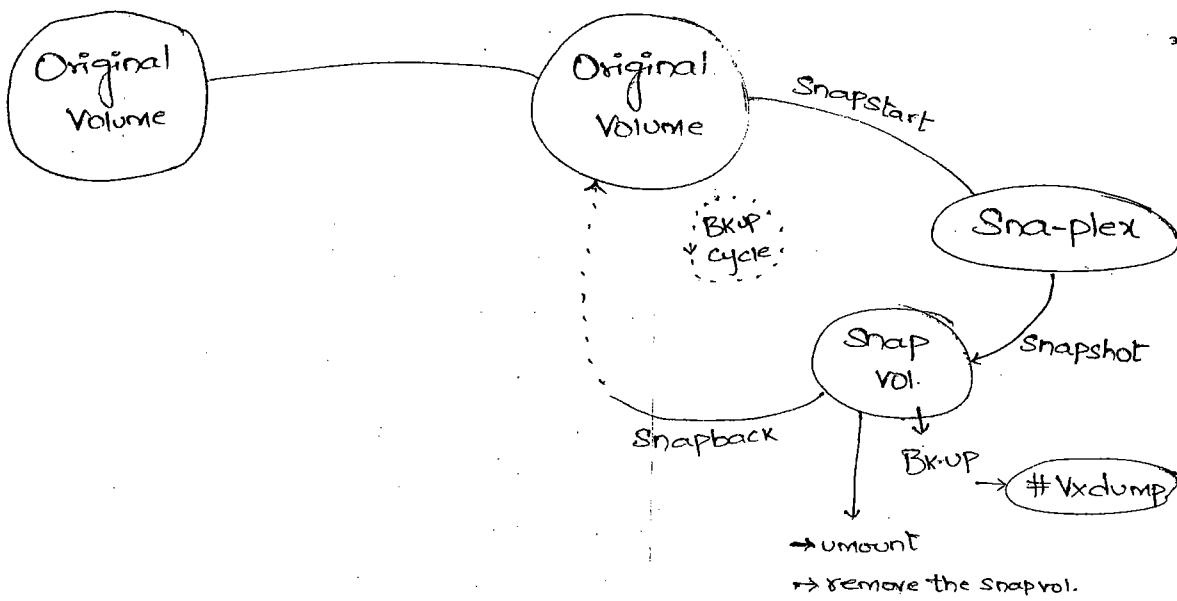
### Drawbacks of Traditional filesystem Snapshots :-

- We Can take Snapshot on filesystem. We Cannot take Multiple Snapshots on filesystem.

- Always UFS Snapshots are Read-only.

\* Mainly Used for Performing the online backups.

\* We can Create Multiple Snapshots



## Commands:

```
# Vxassist -g Sundg Snapstart Convol
```

↓  
volname

```
# Vxassist -g Sundg Snapshot Convol
```

# mkdir /test

# Mount it

# Then take backup of Vxdump

# Unmount it after taking backup.

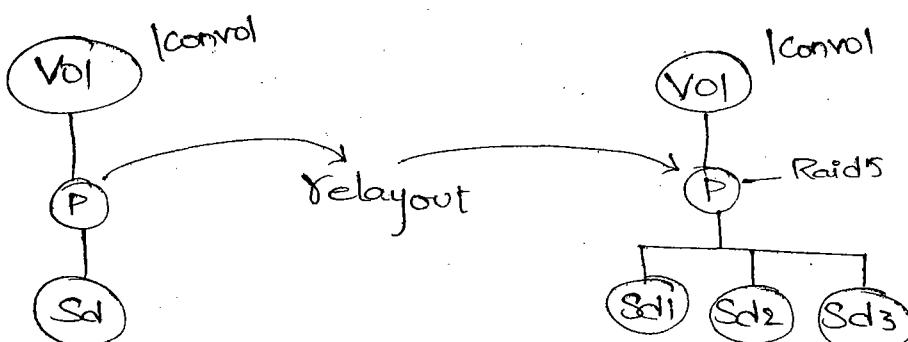
```
# Vxassist -g Sundg Snapback SNAP-Convol
```

(or)

Remove Snap-Convol

```
# Vx edit -g Sundg -rf &m SNAP-Convol.
```

## Relayout's : (online Relayout's)



## Commands:

Concat → Raid5

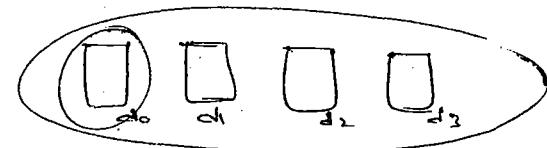
```
#Vxassist -g Sundg Vlayout Conv1 layout=Raid5, noraidslc
```

↓  
(existing vol)

Swidth=64 NCol=3,

#Vaprint -hvst Conv1 ↴

Relocations :- ("to hotsparepool")



```
#Vxedit -g Sundg Set Reserve=on do ↴
```

```
#Vxdg -g Sundg Reserve ↴ : displays how many disks are reserved.
```

→ By default in VxVM each & every disk will act as hotspare disk.

```
#Vxedit -g Sundg Set nohotuse=off on ↴
```

: By executing this cmd. the disk d1 will not participate in the hotspare disks. (i.e., excluding the disk d1 from hotspare disks).

## Encapsulation:-

is a process of bringing the traditional Operating System disk to Veritas Volume Manager.

- generally we need to use this option for Mirroring.

## Pre-Requirements for Performing the Mirroring:

- Min. 2 slices should be free
- Root filesystem should not increase more than 90%.

### Note:

The disk we have to initialise the traditional disk under Sliced format only.

## Command for Encapsulating a disk:-

```
#Vxencap -c -f Sliced -g Rootdg do=Catodo
```

↓            ↓  
Create      Specify  
dg          the format

```
#Vxdisk list
```

Before encapsulating a disk we need to take some precautions.

- We need to take all main filesystem backup.
- We need to take some commands backup.

letc|fstab

letc|system

df -h

letclmnttab

#df -h

If changes are not made then Reboot (init 6)

# VxdiskSetup -i C0t1d0 format=sliced

# Vxdg -g rootdg adddisk di=C0t1d0

# Vxmirror -g rootdg do di f

# Sync

# init 0

OK boot do

# PrtConf -vp | grep boot : displays Present boot disk.

Difference b/w Vxencap & VxdiskSetup

# Vxdisk -o alldgs list

(ordg)

(ord...

: displays all the disks available under Veritas volume Mgr.

# Vxdisk add

: to add

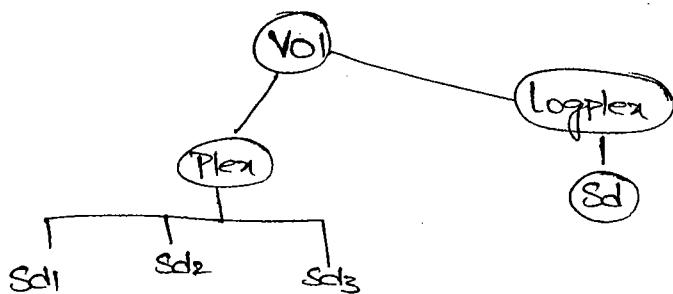
initialize 2 or more disks at a time.

# Vxdisk adm

: Veritas administration at Command level

Raid-5 log :-

# Vxassist -g Sundy Make xsvol 300M layout=Raid5

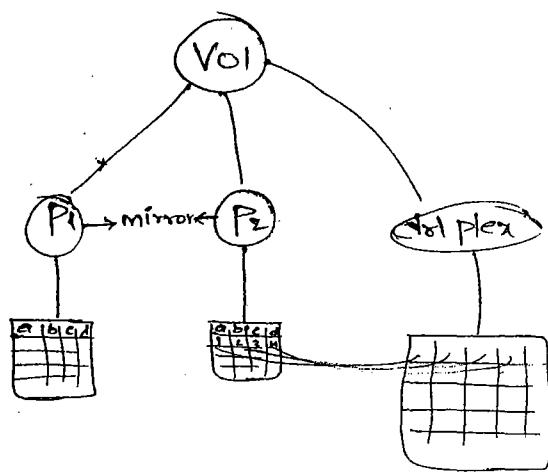


- generally Logplex is used for reduce the synchronization time.

If we manually dissociate Sd1 from Plex then after the dissociation what all the I/O operations performed will be stored in Sd2 & Sd3. These newly added data index will be maintained by Logplex in Sd. After that when we attach the Sd1 to Plex then it will get the index from Logplex & only the newly added data after dissociation will be synchronised in Sd1.

If entire Sd1 is corrupted then we can't do anything have to sync. entire data.

## Dirty Region - Logins :-



```
#Vxassist -g Sundy Make mirror  
Zoom layout=mirror  
logtype=drl ↵
```

- used to copy the newly added data.
- to reduce the synchronization time.

When we Create a mirrored volume & Perform some I/O operation after some time we disassociate P1 then after dis the I/O operations performed will be stored under P2 & drl will maintain the index.

## How to dedicate a drplex to an existing volume.

```
#Vxassist edit -g Sundy Mirvol addlog logtype=drl ↵
```

Demo :-

## Snapshot

# Vxassist -g Sundy make Convol zoom

```
# mkfs -f vxfs /dev/vx1r0dsk1s0d1g1c0n0
```

# Mkdir | Concat

```
# Mount -f vfat |dev|ra|disk|sundg|control|concat
```

+ cd | Concat

```
# touch ab cd ef <J
```

#Vrassist -g Sundg Snapstart Conrol ↴

#Vaassist -g Sunday Snapshot Control ↪

# Vapoint - hrvpst

Create dir. & Mount it (don't build fs)

# MKdir Testvol

# Mount -F Vafs [dev|vx|disk|sundy] snap-control | testroot

# 0d |testv0|

井 18

Taking backup of Veritas pls:

# Vxdump -ocuf Nestvol. dump | testvol ↴

<target file>

```
# Vx Restore -tf /testvol.dump.
```

```
# Umount /testvol
```

```
# Vxassist -g sundg Snapback SNAP-control
```



Here in Veritas Volume Manager the Snapshot is  
readonly. after mounting it we can write  
data.

Where as in Traditional file system only readonly.

## Relayouts

Create Concat Vol & build files.

```
# Cd /Concat
```

```
# Cp /etc/*
```

to change from raid0 → raid5

```
# Vxassist -g Sundg Relayout Control layout=raids, no  
raidslog Swidth=61
```

Ncd = 3.

```
# iostat -cndc
```

## Relocations:

Reserve the disk.

#Vredit -g Sundg set Reserve=on do ↴

Vadisk list

#Vxassist -g Sundg Make Control Zoom.

#Vapoint -hvpt

#Vredit -g Sundg Reserve=off do ↴

## hotspare

#Vredit -g Sundg Set nohotuse=on do  
:on all hotspare

## Encapsulation :-

```
# Vxencap -c -f Sliced -g rootdg do=cotodo k
```

```
# Cp /etc/vfstab /etc/vfstab.bkp
```

```
# Cp /etc/system /etc/system.bkp
```

```
# Cp /etc/mnttab /etc/mnttab.bkp
```

```
# df -h >/df.bkp
```

```
# sync ↴
```

```
# init 6 ↴
```

```
# Vxdisk list ↴
```

```
# Cat /etc/vfstab ↴
```

```
# df -h
```

```
# Vxdisksetup -i cotido format=Sliced.
```

```
# Vxdg -g rootdg adddisk di=cotido ↴
```

```
# Vxmirror -g rootdg do di ↴
```

```
# Vxtask monitor.
```

```
# init 0 ↴
```

```
ok> boot do ↴
```

```
# PktConf -vp | grep boot ↴
```

## Lab Demo: CJS (Custom Jump Start)

- We can install Operating System in multiple ~~systems~~ through Network.

### Needs

Client System MAC Address i.e., ethernet address (Physical address).

1. Create Slice with 5Gb & Mount it (cjs)

- Insert 1/4 cd It should mount automatically.

```
# cd /cdrom/cdromo/Sol/Solaris-10/Tools ↵
```

```
# ./Setup-install-server cjs ↵
```

```
# eject ↵
```

- Insert 2/4 cd

```
# cd /cdrom/cdromo/Sol/Solaris-10/Tools ↵
```

```
# ./add-to-install-server cjs ↵
```

```
# eject ↵
```

Create one More Slice with 1gb & Mount it (config)

```
# cd /Config ↵ (for creating Client Profile files).
```

```
# touch Profiles Rules sysidcfg ↵
```

Rules - for maintaining system rules.

```
# vi Rules ↵
```

any -- Profile --  
↓ ↓  
anywhere starting point ending point ↵

# Vi Profile : to maintain the client side parameters

install-type initial-install

System-type Standalone

locale en-US

geo Asia/Calcutta

Cluster SUNxcall

Partitioning explicit

filesys C:\ 2000\

filesys C:\ 1024 Swap

filesys C:\ 6000\usr

filesys C:\ 3000\var

filesys C:\ 5000\opt

:wq!

# Vi Sysidcfg : to maintain the network information of Client

System-locale=en-US

Security-Policy=none

Name-Service=none

Network-interface=Primary {Protocol\_IPv6=no Netmask=255.255.255.0 default-route=none}

timezone=EST

timeserver=localhost

terminal=vt100

root - Password = KUC580W1...

(for root password goto /etc/shadow & copy the 2nd field and Paste it here) It will be in encrypted format.).

:wq!

# df -h

# cd /cjs

# ls

# cd Solaris-10/Misc/Jumpstart-Sample

# ls

# cp -r check /Config : for checking Configuration of cjs

# cd /Config

# ls

# ./check

# ls

# vi /etc/dfs/dfstab

Share -f nfs -o ro,anon=0 /cjs

Share -F nfs -o ro,anon=0 /Config

:wq!

# Shareall

# Share

# Cd /etc

# vi ethers : to create Mac id of client & hostname.

to get macid ok> bootbanner.

8:0:20:c8:db:bl rootsun21

:wq!

# vi hosts

Lastline  
200.200.0.21 wostsun21

:wq

# cd root

# cd |cjs|Solaris-10|Tools|

# ls

# ./add-install-client -c wostsun6:|cjs -P wostsun6:|config  
↓  
Configuring

Updating /etc/bootparams.

wostsun21 < Sun4u <

↑  
in intel (i86pc)

## CJS Server daemons:

### Serverside daemons

in.rarpd (reverse address protocol)

in.tftpd (trivial file transfer protocol)

### Client side daemon

arp (address resolution protocol)

## Client sys

ok>boot net - install

7/11/09

## CLUSTER :-

What is Cluster?

- It's a Collection of Systems/nodes (or) Providing on Application Services with high availability & scalability.
- Generally Clusters are used to Provide 24x7 Uptime for Servers.

& we have different types of clusters

Sun → Suncluster

HP → Service guard

IBM → H.A.C.M.P (High Availability cluster Multipathing)

Symantic → V.C.S (Veritas Clusters)

- VCS is an 3<sup>rd</sup> Party tool. It can be used on any Platform.

## Veritas Cluster:

- It's a Combination of two (or) More Systems (or) nodes.
- VCS is an 3<sup>rd</sup> Party appn sw which is Provided by the Symantic organization.
- Package for Veritas Cluster VRTSVcs

- all the nodes in a cluster should maintain same OS, same hardware

- at Max. We can configure 32 nodes in a one single cluster environment.

- Min. 2 node cluster, Max. 32 node cluster.

- We can configure a Veritas cluster in a 2 ways

(i) Graphical User Interface (G.U.I) #hagui &

(ii) Command Line Interface (C.L.I)

(i) # hagui (high availability graphical user interface).

- used to configure clusters in GUI mode.

(ii) C.L.I → 

|                    |
|--------------------|
| # hagrps <options> |
| # haconf <options> |
| # haress <options> |
| # hauser <options> |
| # hasys <options>  |

- VCS provide the services with the help of Service group

### Terminology in VCS:

- Service group

- faulted

- Switch over.

- Resources & Resource dependency's

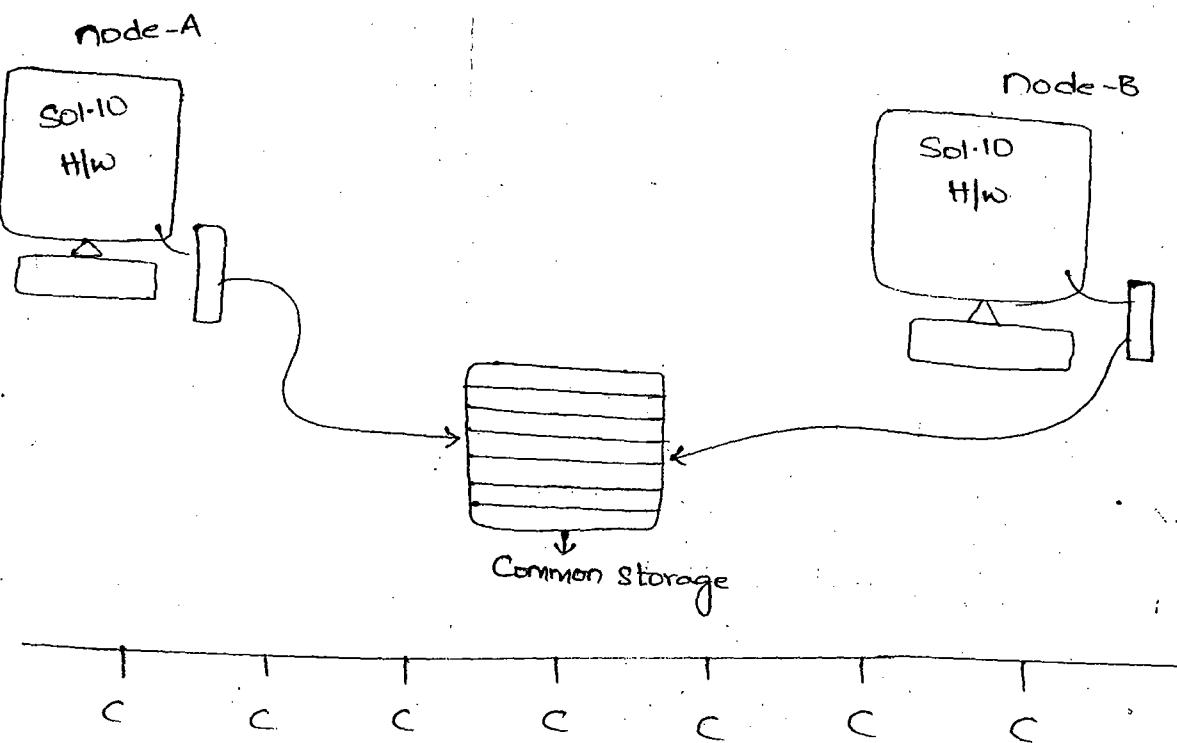
- Agents

- Protocol

- Public Communication

- Private Communication

## Basic Cluster (2 node Cluster) :-



If 1 node goes down another node is comes in to Picture.  
Clients are dynamically connected to other node. The concept of 1 node going down is called "failover".

## Service group:-

- is a combination of resources & resource dependencies.

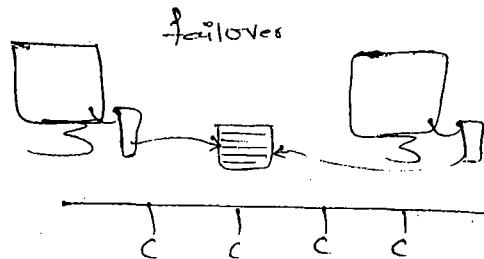
- Vcs can provide the services by creating Service group.

## Types of Service group:

1. failover
2. Parallel
3. hybrid

- Based on the Configuration & availability they are introducing different types of service groups.

- 1. failover:
    - Basic Service group
    - we need to configure in b/w 2 nodes.



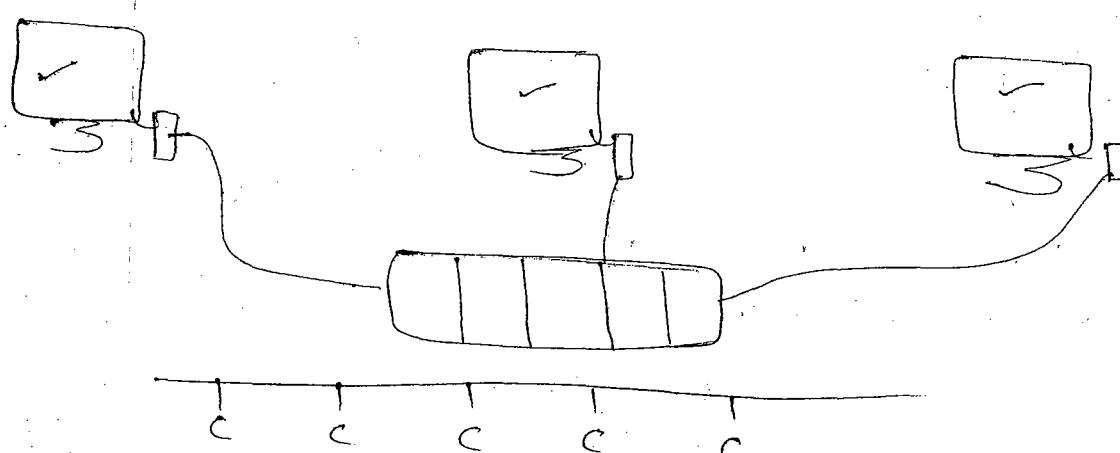
- In a failover Service group both nodes are upend running but only one single node can provide the resources to clients.

-If one node goes down then automatically another node will come up.

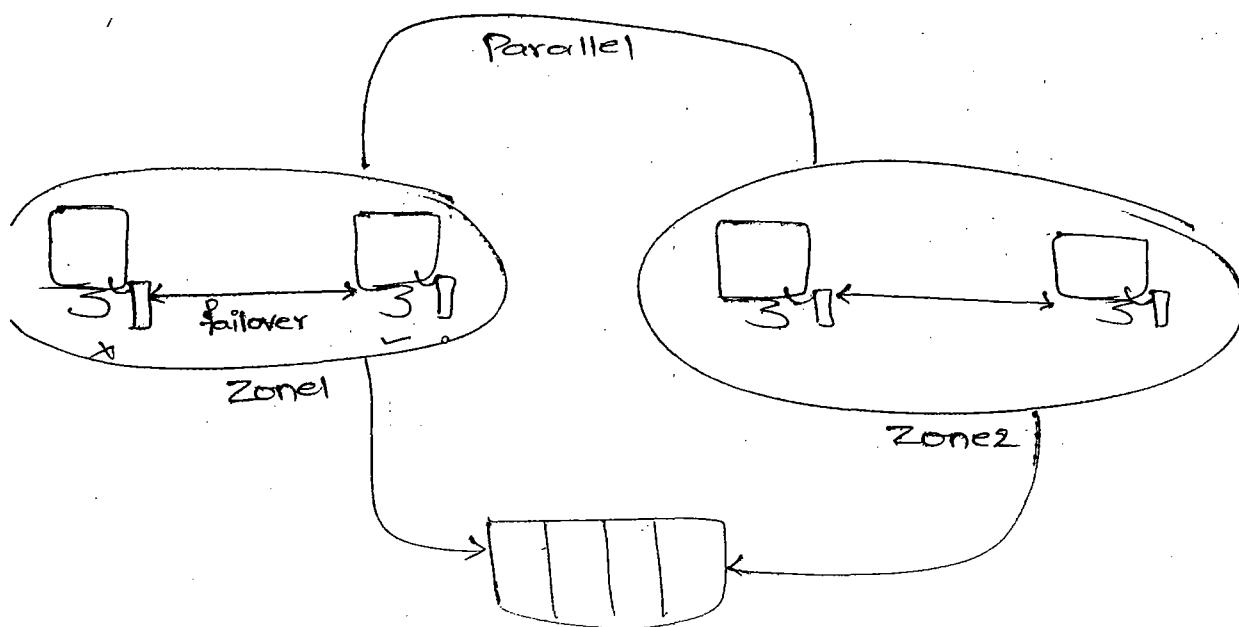
- In failover service group there is no load balancing.

2. Parallel: - By Configuring the Parallel Service group We can share the resources to multiple nodes.

  - Load Balancing is high.
  - Min. is 3 Systems need to config. Parallel Service group.



3. Hybrid: - It's A Combination of Parallel & failover Service group.
- Min. 4 nodes are needed for Configuring the hybrid cluster.
  - It Provides high availability & Performance.



### Service group States:

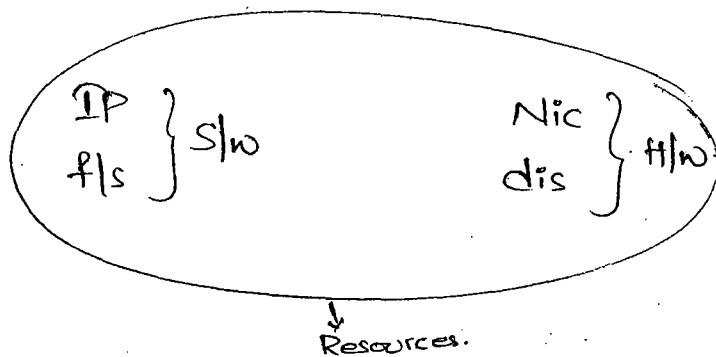
\* online: A Service group Said to be online When the resources and resource dependencies are active & running.

\* offline: A Service group Can Said to be offline When it is temporarily Shutdown.

\* faulted: A Service group in faulted state when there is any faults in System like -due to improper resources (or) due to improper Configuration files.

Resources: - Resource is nothing but it's an entity, it may be SW or HW.

- Resources are used to Configure the Service group.



### Types:

1. on-only
2. On-Off
3. Persistent

1. on-only Resources:- The Resource which going to be made online by the VCS but we cannot make offline by the VCS.

Eg: daemons      VCS daemons we cannot make offline.

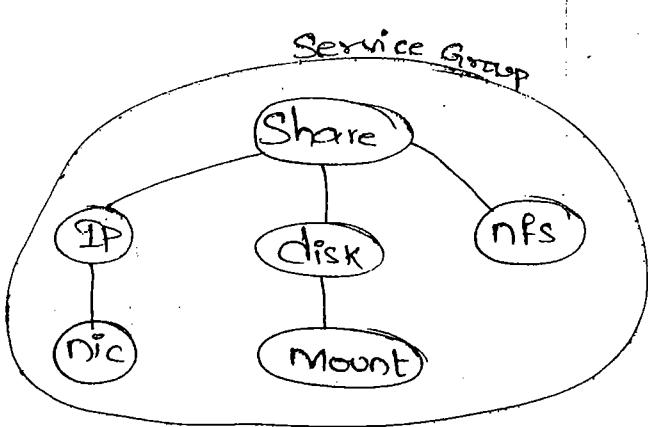
2. On-Off Resources:- The Resource which we can bring online as well as offline by the VCS.

Eg: IP, filesystems, Mountpoints.

3. Persistent: The Resources which cannot be brought online/offline Manually.

Eg: Nic, disk

## Resources & Resource dependencies hierarchies:



Combination of Resources & Resource dependencies is called a Service group.

Agent's:- - Agents are nothing but Vcs Process.

- The agents are defined to monitor the resources
- We may have different types of agents in a Cluster Environment.

### Types of Agents:-

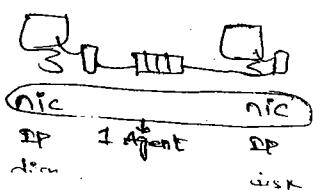
1. Bundled

2. Enterprise

3. Custom

1. Bundled Agent: The Agents which are inbuilt in a Cluster SW those type of agents are called Bundled Agents

Eg: NIC Agent, Disk Agent, IP Agent, NFS Agent



2. Enterprise Agents: - The Agents which we can download for Enterprise Agents.

Eg:- Vnb, Oracle

3. Custom Agents: The Agents which are going to be build Manually depending upon the Client Request.

Protocol:

\* Gab

\* llt

\* Gab : (Global atomic broadcast)

- Gab Protocol is responsible for Maintain the Cluster Membership.

- It is always responsible for sending the Signals to llt Protocol.

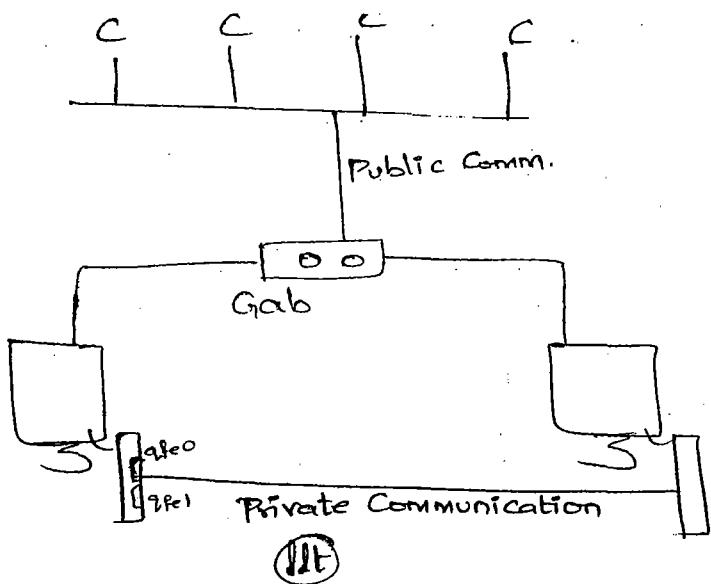
- Responsible for Maintain Public Communication.

\* llt : (Low latency Transfer Protocol)

- This Protocol is responsible for Transfer Every cluster status from one node to another node.

- It is responsible for Maintain Private Communication b/w the nodes.

- It regularly checks heart beats from one node to another node.



If two nodes try to access one Single Storage then

**Splitbrain** occurs

↓  
Flooding

### Deamons:

- \* **HAD** :- The HAD deamon is responsible for maintaining the config. Info. in all nodes.
  - HAD deamon is responsible for update "Gab" Protocol.
  - HAD (High Availability deamon).

- \* **HAD - Shadow** :- This deamon is responsible for restart the HAD deamon. in case of HAD deamon fails.

- HAD deamon is responsible for reading & updating 2 main Configuration files

**main.cf**

& **types.cf**

10/11/09

## Minimum Requirements for Installing the VCS :-

- Create /opt filesystem with 3gb.
- Install Sol.10 OS on both nodes. Provide demo license key.
- download the SW from CJS Server.

### Demo Clusters :

(hostsun16, hostsun17)

#### hostsun17:

```
# Ping 200.200.0.22
```

```
# ShowMount -e 200.200.0.22
```

```
# Mkdir /test
```

```
# Mount -f nfs 200.200.0.22:/opt /test
```

```
# df -h
```

```
# cd /test
```

```
# cd sfhun1.1-s1
```

```
# cp -r sf-hacn1 /opt
```

another console

```
# du -sh /opt
```

```
# umount /test ↴
```

```
# cd /opt ↴ ls ↴
```

```
# file sf-hacn1
```

```
# tar -xvf sf-hacn1 ↴
```

Maintain /etc/hosts file on both nodes with both nodes host names & IP address.

Establish the trusted relationship on both nodes.

# vi /etc/hosts

200.200.0.16 wostsun16

200.200.0.17 wostsun17

:wq!

# Cat >/etc/hosts

+

Ctrl+d

\* to maintain the trusted relationship we need to create the .hosts file in the root directory.

# Ping 200.200.0.16

# Ping wostsun16.

wostsun16:

# vi /etc/hosts

200.200.0.17 wostsun17

200.200.0.16 wostsun16

:wq!

# Cat >/etc/hosts

+

Ctrl+d

# Ping 200.200.0.17

```
#> login wstsun17
```

if it comes without asking Password then  
the trusted relationship is done.

```
#ctrl +d ↵
```

Check the date on both nodes.

wstsun17:

```
# cd /opt
```

```
# ls
```

```
# ./installer ↵
```

:T

:1

Enter the system names: wstsun17 wstsun16

↓  
Infrastructure PKgs.  
↓

Vcs License Key : ipaq-perq-std8-yglv-80ph-8ccc

(n) ↵

(n) ↵

↓

: 1 ↵

↓

Are you ready to Configure Vcs : y ↵

Enter Unique Cluster name : vcs

Cluster id no. : 0

Enter the NIC for Private heartbeat link on westsun17 : eth0

At max we can provide 8 private communication.

Would you like to Configure second Private heartbeat link : eth1

3rd

: no ↴

Do you want to Configure an additional low priority heart  
(Public Comm.)  
beatlink : y ↴

Enter .... : (homeo)

Are you using same NIC's : yes ↴

VCS Security Services : No ↴

By default Username is : admin Password : Password

(a)

We can also Configure.

y ↴

Enter user name : raj

Password : ↴

Is this info Correct : ↴

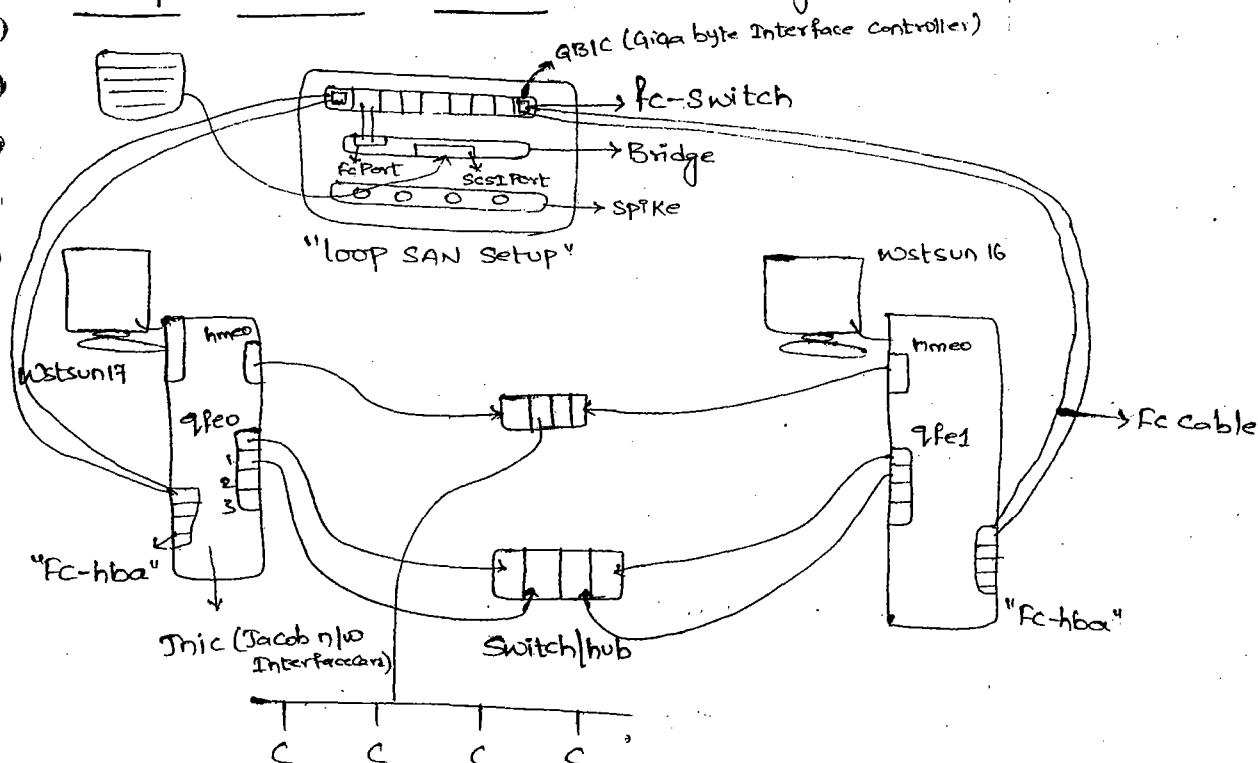
want to

Do you want to Configure SMTP notification : n ↴

SNMP : n ↴

Install Cluster Services on all : y ↴

## Loop SAN Setup:- Making Public & Private Comm.



Bridge is used to Convert SCSI Signals to fc Signals & FC Signals to SCSI Signals.

Starting Cluster Server.

Installation log for VCS

OPT|VRTS|install|logs

When we install VCS it updates 2 locations.

↳ to see what exactly happens at installation

letc|VRTSvcs : Cluster related Main Config. file.

cd letc|VRTSvcs|Conf|Config|

Main.cf  
Types.cf

|\opt|VRTS|bin|: Responsible for maintaining the Commands

# vi /etc/profile

lastline

PATH=\$PATH:/opt/VRTS/bin

export PATH

:wq!

# . /etc/profile & #sh /etc/profile

# ha<tab><tab>

# Pkginfo |grep VRTSVCS

# modinfo |grep -i vcs

Configuring the VCS:

In GUI

# hagui

Click on: New cluster

Hostname : westsun17

Port : 14141 (default Port no. for vcs)

failover Retries : 12

Username : say

Password : ↴

Service groups:  No ↴

## Configuring / Creating the Service group:

Goto **Tools** → **Configuration Wizard**

Switch to Read/Write mode  Yes ↴

**Next** ↴

Service group Name: Sunsg ↴

Select the hostname & add. : it is a Priority.

Service group . ⚡ failover.

**Next** ↴ **Finish** ↴

## Creating Resources:

Right click on Service group (Sunsg)

**Add Resource**

Resource name : nicyes

Resource type : NIC

Select bold one

Device String Scalar

Edit ↴

global  
 Persystem

|              |
|--------------|
| Scalar value |
| homeo        |

[OK] ↵

✓ Enabled

Click [OK]

### Creating IP Resource:

Right click → Add Resource

Resource Name: IPres

Resource Type: IP

Edit the bold ones.

Device

edit

Address

|                                         |              |
|-----------------------------------------|--------------|
| Device                                  | Scalar value |
| <input checked="" type="radio"/> global | homeo        |

[OK] ↵

Address

Global

|               |
|---------------|
| Scalar value  |
| 200.200.0.169 |

! no need to create virtual IP Manually Yes automatically creates.

## Creating Disk Resource:

Right Click → addresource.

Resource name : diskres

Type : Disk

Edit bold ones.

Create the Partition & build f/s on external Storage

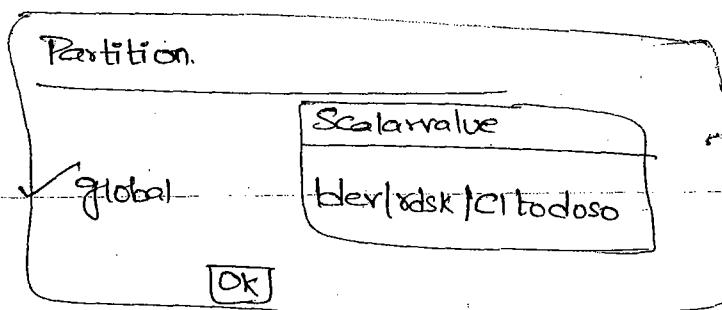
#format

Don't Mount it

#mkdir /sun ↴

#rsh root@sun16 mkdir /sun ↴

Edit bold ones.



✓ Enabled

OK ↴

## Creating Mount Resource:

Rightclick on Servicegroup → add resource.

Name: Mountres

Type: Mount

Edit bold ones.

Mountpoint

global

Scalarvalue  
/sun

OK

: has to specify absolute path

Block device

global

Scalarvalue  
/dev/dsk/c1t0d0s0

OK

FSType

global

Scalarvalue  
UFS

OK

FSCK OPT

global

Scalarvalue  
-y

OK

Enabled

OK

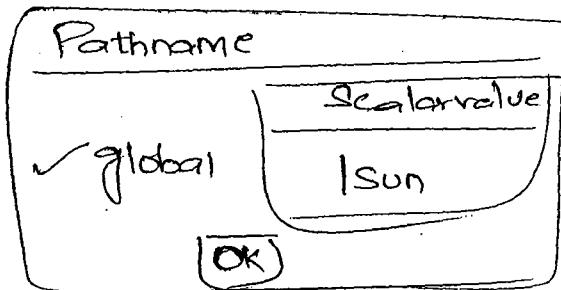
## Creating Shared Resource:

Rightclick → add resource.

ResourceName : Shareres

Type : Share

Edit bold ones.



✓ Enabled

[OK] ↳

## Creating NFS Resource:

Have to delete NFS service in System

# Svccfg delete nfs/server

# rsh rootson16 Svccfg delete nfs/server

Right click → add Resource

Resource name : nfsres

Type : NFS

✓ Enabled

[OK] ↳

After Creating Resources.

Making dependency hierarchy.

On top go to [Resources] tab

Rightclick on Share res. → link

Select : IP res

[OK]

Rightclick IP res → link

Select: nic res

[OK]

Rightclick : Share res → link

Select : Mount res.

[OK]

Rightclick Mountres → link

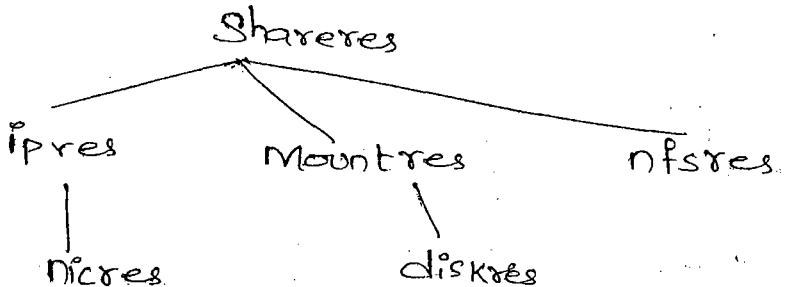
Select : diskres.

[OK]

Rightclick Share res → link

Select: NFS res

[OK]



Enable the resources.

Right click Service group → Enable Resources.

Right click Service group → online → wstsun17

Are you sure [Yes] ↴

How to Switch the Service group from one node to another node.

Rightclick Service group → Switch to → wstsun16

Are you sure [Yes] ↴

Console:

wstsun17

ifconfig -a  
df -h

Console

wstsun16

ifconfig -a  
df -h

Command Level :

# haConf -makeRW : make Config files RW.

Creating Service group.

• State means RW

# hagrp -add Sunsg2 ↴

Add Systems

# haupd -modofin Sunsg2 SystemList nbtssunit 0 wstsun16 1

#hagrp -modify Sunsg2 AutoStartList Wstsun77 Wstsunk2

#hagrp -modify Sunsg2 Parallel 0 ↴

# hagrp -list : used to display Service groups.

# hagrp -display Sunsg2 : detailed info.

### Creating Resources:

#### 1. Creating NIC Resource

#hares -add nicres2 NIC Sunsg2 ↴

#hares -modify nicres2 Device hme0 ↴

#hares -modify nicres2 Critical 1 ↴

#hares -modify nicres2 Enabled 0 ↴

#hares -list ↴

#Ptdiag -rl prep  
#dadm

#### 2. Creating IP Resource

#hares -add ipres2 IP Sunsg2 ↴

#hares -modify ipres2 Device hme0 ↴

#hares -modify ipres2 Address 200.200.0.148 ↴

#hares -modify ipres2 Critical 1 ↴

#hares -modify ipres2 Enabled 0 ↴

#hares -list ↴

### 3. Creating Disk Resource:

Create a Partition & build filesystem.

don't Mount but Create Mountpoint

#mkdir /oraeng ↴

#rsh wstsun16 mkdir /oraeng ↴

#hares -add diskres2 Disk Sunsg2

#hares -modify diskres2 Partition /dev/rdsk/c1t0d0s1 ↴

#hares -modify diskres2 Critical 1

#hares -modify diskres2 Enabled 0

#hares -list ↴

### 4. Creating Mount Resource:

#hares -add mountres2 Mount Sunsg2

#hares -modify mountres2 MountPoint /oraeng

#hares -modify mountres2 BlockDevice /dev/ldsk/c1t0d0

#hares -modify mountres2 FSType UFS

#hares -modify mountres2 FsckOpt -y ↴

#hares -modify mountres2 Critical 1 ↴

#hares -modify mountres2 Enabled 0 ↴

#hares -list ↴

## 5. Creating Share Resource.

```
#shares -add shareres2 Share Sunsg2 ↴  
#shares -Modify shareres2 PathName lobeng ↴  
#shares -Modify shareres2 Critical 1 ↴  
#shares -Modify shareres2 Enabled 0 ↴  
#shares -list ↴
```

## 6. Creating NFS Resource.

```
#shares -add nfsres2 NFS Sunsg2  
#shares -modify nfsres2 Critical 1  
#shares -modify nfsres2 Enabled 0  
#shares -list ↴
```

## Linking/Making dependency hierarchy.

```
#shares -link shareres2 ipres2  
#shares -link ipres2 nicres2  
#shares -link shareres2 Mountres2  
#shares -link Mountres2 diskres2  
#shares -link shareres2 nfsres2
```

| Enable the Resources & Make online.

| #hagr -enable resources Sunsg2

| #hagr -online Sunsg2 -sys westsun17

| #df -h

| How to Switch over Service group from one system  
to another!

| #hagr -Switch Sunsg2 -to Westsun16

| #df -h

| #rsh Westsun16 df -h

| #rsh Westsun16 ifconfig -a

| **#haStatus -sum**

: Used to display on which Machine Service group is online.

| **#Cat /etc/llt hosts**

: how to see how many nodes are participating in a cluster.

| **#Cat /etc/llttab**

: We can able to see Public & Private Communications.

| **#Cat /etc/gabtab**

| **#/sbin/gabConfig () -c n2**

: Make the Config. Entries.

```
# cd /etc/clusterd ↓  
# ls  
# ./sqzgab stop ↓  
# ./sqzgab start ↓  
# ./stollt stop ↓  
# ./stollt start ↓
```

How to add another user to a cluster:

```
# hauser -add <username> -priv Administrator.  
↓  
to specify administrative perm.
```

```
# hauser -list
```

```
# hauser -display
```

To delete a user.

```
# hauser
```

11/11/09

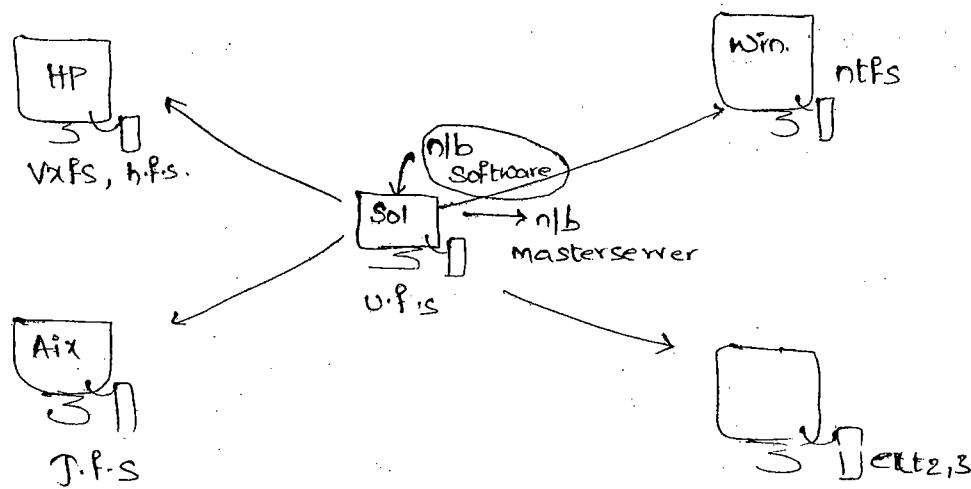
## VERITAS NET BACKUP

### Drawbacks in Ufsdump :-

- We can't Perform heterogenous platform backup's.
- we Cannot take Multi terabyte filesystem backup.
- Synthetic backup is not Possible.
- It won't Provide Centralised backup info.

### Net backup Advantages:-

- It can Support heterogenous platform backup & restore.
- Capable of taking backup of terabyte of data.
- It maintains the backup info. in the form of Catalog's.  
(meta db)
- Scheduled backups can be done.
- Provides GUI for backup & restore.
- User's also can initiate the backup if they have Privileges.



## Tools for Net backup:-

HP → omni

IBM → Tivoli

Symantic → V.n.b  
(3<sup>rd</sup> party)

}

These tools used to take backup.

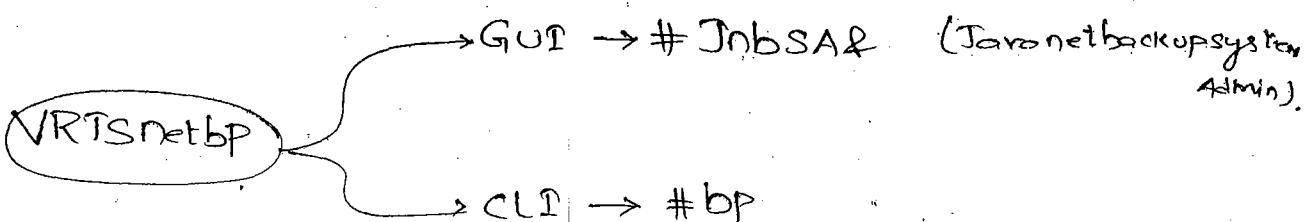
## Veritas Net Backup:-

- Vnb is a 3<sup>rd</sup> Party SW appn provided by the Symantic organization.

Package → VRTSnetbp

- We can Configure Vnb in 2 ways.

- G.U.I (Graphical User Interface)
- C.L.I (Commandline)



## Versions:

|     |     |          |
|-----|-----|----------|
| 3.0 | 5.0 | → lab    |
| 3.1 | 5.1 |          |
| 4.0 | 6.0 | → latest |
| 4.1 |     |          |
| 4.5 |     |          |

## Net backup Components:-

→ Master Server

→ Media Server

→ Client.

→ Master Server :- -The actual SW installed on Netbackup Master Server only.

- Master Server is Responsible for laying the Communication b/w Media Servers & Client

- Total Administration Can done on Master Server only.

- Master Server is Responsible for Maintain 2(Or) More Media Server.

## Media Server :-

- The Actual Storage is directly Connected to Media Server.

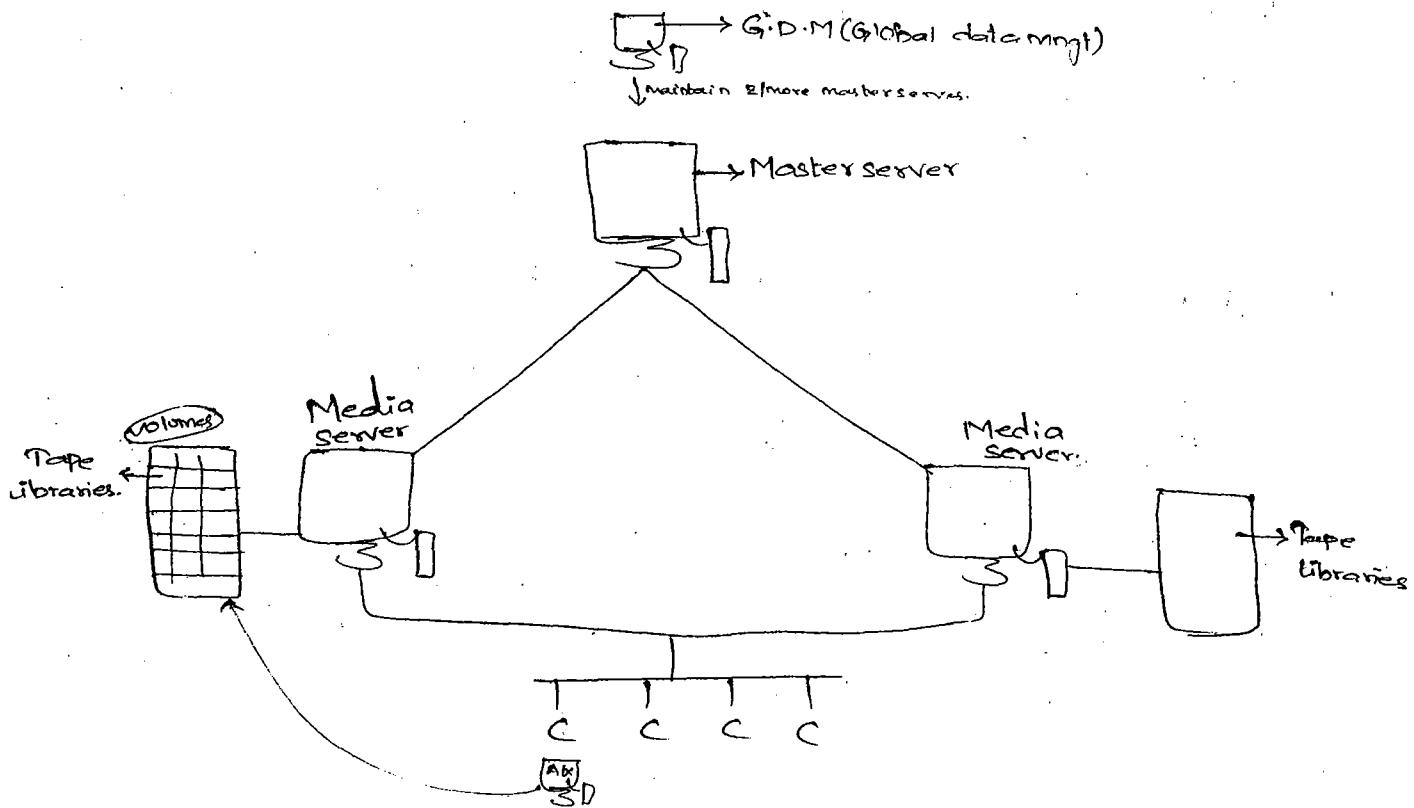
- It is Responsible for Providing the backup Environment for Netbackup clients.

- Load Sharing Can be done on Media Servers.

- It is Responsible for Provide Volumes(Or) tape Libraries.

→ Client :-

- Client May be any System.



Backup Levels :-

Incremental backup (day by day)

Cumulative backup (collection of incrementals)

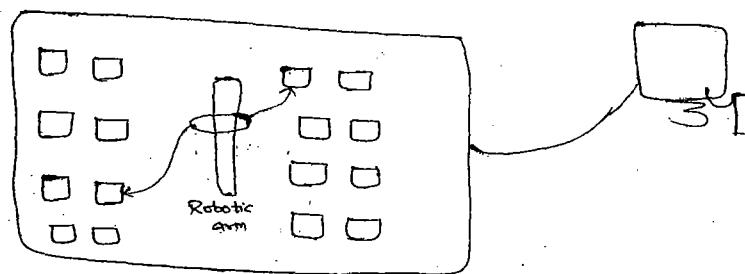
Full backup

Differential backup (difference b/w two Incrementals)

User backup (#.bp) — As a client / user also we can take backup with the help of #.bp . If he is netbackup client only.

## Auto Loaders :-

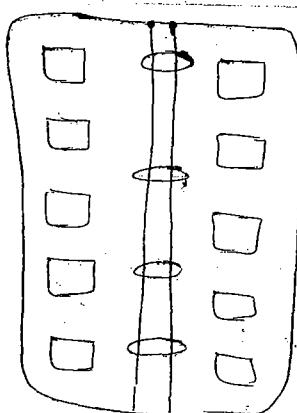
- used to take backup of two (or) More clients.
- Auto loaders are not Scalable
- In a Single Autoloader we have a single Robotic arm



- we have barcodes on each & every tape drive.
- looking at that barcode robotic arm can <sup>know</sup> whether the cartridge is empty (or) not.

## Tape library :-

- Collection of 2 (or) More tapes.
- It can have 'n' no. of robotic arms.
- Tape libraries are Scalable.



SL-8500

## Net backup Terminology:

### Volume Pool :-

— Collection of 2/more tape drives. (Cartridges  
(or) Volumes)

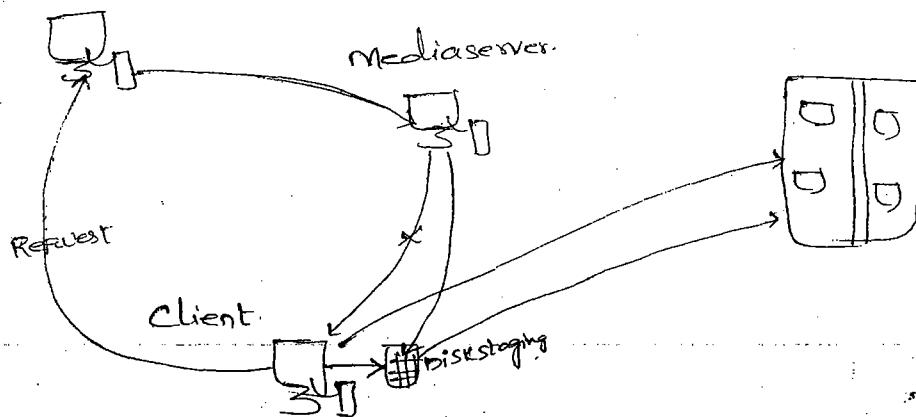
Storage unit:- — One (or) more devices which are directly connected to the Media Servers are called Storage units.

Media Manager Storage unit → It handles tapes & tape libraries.

Disk Storage unit → Responsible to store the backup info. in local internal (or) external disk.

### Disk Staging :-

Masterserver



When the Client Sends backup request to the Masterserver. Then it sends signal to media server & media server will take the backup request from that particular client & stores in tape drives.

Now, if at all the MediaServer is busy at that time of Client request. Then Client Needs to store

the information in his local System on external disks.

After some time when the Media Server is free then it automatically search for the backup info in the Client external disk and stores in the tape libraries.

This is Disk staging.

### Two Steps:

- Store the data in internal/external disks.
- After storing in external disks, Media Server is responsible to store the data in tape library.

### Policy:

- Policies are used to define 'WH' type of questions

- When to take backup?
- Where?
- What?

1 Policy → 2/more clients

1 client can participate in 2/more policies.

### Catalog:

- is a db which is going to maintain vital info. of the backups.

- it is similar to Metadatabase.

- we can store catalog bup info. hard drives or tape libraries.

## Barcode:

- generated by Vendor & Placed under the Volume.

- Media Servers Uses this barcode to identify the tapes (or) volumes.

## Deamons for VNB :-

### \* bprd : (backup request deamon)

- Responsible for taking the clients backup request.
- It resides on Master Server.

### \* bpdld : (backup database deamon)

- Responsible for Maintain the Catalog database backups.
- Resides on Masterserver & Media servers.

### \* Vnd : (volume Mgt deamon)

- Responsible for Maintain. Volumes info.

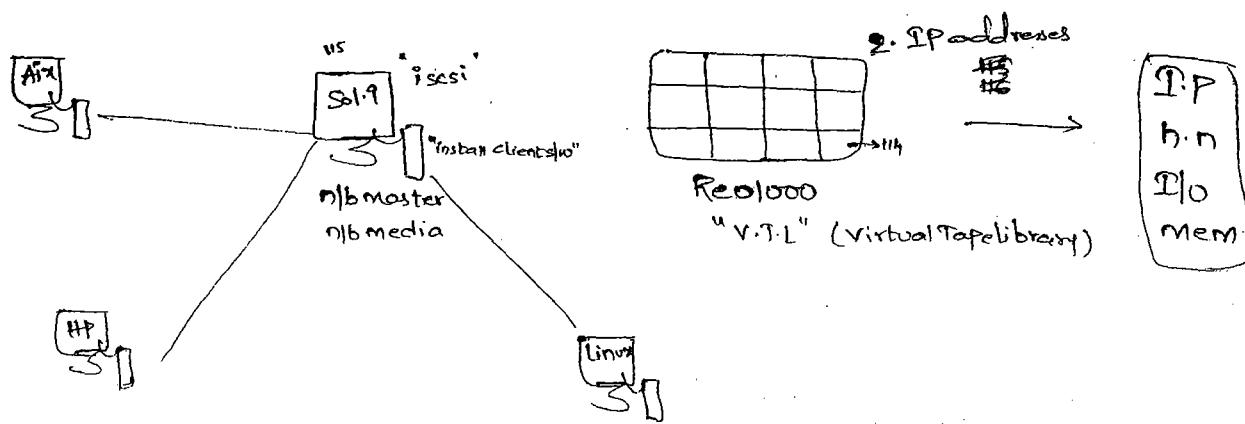
### \* bpcd :- (backup client deamon)

- Responsible for Maintain the communication b/w netbackup Master server & netbackup Client.

12/11/09

## Minimum Requirements for Installing the VNB SW:

- Create lopt f/s with 3GB
- Install Solaris 9 O/S.
- Install iSCSI Patches (To Communicate with Storage).
- Provide the demo license key.



After Installing iSCSI patches it's Mandatory to Reboot the System.

Before taking b.kup Mandatory to install Client SW.

After Installing Sol.9 O/S.

Net backup Master Server hostname : Vnb.

# ifconfig -a

192.168.100.115

- Download the iSCSI patches from CJs Server.

Note: Don't remove any files.

```
# cd /test  
# ls  
# cp -r iscsipatch /opt  
# cp -r vnbdump /opt  
# umount /test  
# cd /opt  
# cd iscsipatch  
# ls  
# Unzip all the Patches.*  
# unzip 112834-0.zip  
# unzip 112233-1.zip  
# unzip 113277-3.zip  
# file *  
# Patchadd 112834-06  
# Patchadd 112233-12  
# Patchadd 113277-35  
# gunzip Solaris-.gz  
# file Solaris-  
# tar -xvf Solaris-  
# ls  
- - - ciscoiscsi
```

```
# Pkgadd -d . Cscoiscsi
```

```
# modinfo | grep -i iscsi
```

```
# vi /etc/iscsi.conf
```

Line: 31:

Uncomment it

Discovery Address=192.168.100.114 (iscsi Storage address).

:wq! <

```
# vi /etc/initiatorname.iscsi
```

InitiatorName=iqn.1987-05.com.Cisco:01.2127732e8edc

:wq! <

~~#reboot~~

```
# reboot -- -y <
```

```
# Ping 192.168.100.114
```

Sol. 10

#iscsiadm

```
# cd /dev/smt
```

```
# iscsi-ls
```

Download the Software from CJS Server to /opt directory

```
# cd /opt
```

```
# ls
```

```
# cd vnbdump
```

```
# ls
```

```
# date
```

```
# ./install
```

Choose an option : 1 ↴  
running in root envi. : y ↴  
: y ↴

Do you want to load any other NetBackup clients onto the  
Server? : y ↴

Enter choice : 10 ↴  
: y ↴

---

Commands location for ~~netbkp~~, vnb:

[opt|openv|netbackup|bin] : Original file.

[usr|openv|netbackup|bin] : This is Softlink file.

Set the path variable. /etc/profile

PATH=\$PATH:/opt/openv/netbackup/bin

Export PATH

:wq! ↴

# . /etc/profile

---

Enter the license key: dm6 - PP68 - zdjw - kghP - 6oz0 - 4itp.  
- 6ymM - g77h - P ↴

OK: y ↴

Additional keys: n ↴

Would you like to use "vnb" as the Configured Name  
of the Netbackup Server? : y ↴

Is vnb the MasterServer?: y ↴

:n ↴ → (we need to provide  
FQDN if say yes)

Enter which host will store the global device database(vnb):  
y ↴

Do you want to Start Netbackup SW Process: y ↴

Do you want to Create policy: y ↴

Do you want to Start MediaManager device deamon:  
y ↴  
bprd process : y ↴

ltrace | install -trace .624

: Can Trace the Install Process

# Pkginfo | grep VRTSnetbp ↴

# Modinfo | grep -i net ↴

# Sh /etc/profile ↴

# . /etc/profile ↴

## Configuration in Gui Mode. (Master Server.)

#Jnbsa & ↴

Hostname : Vnb

: root

Password :

Select  Getting Started

Next ↴

Step 1: Configure Storage devices.

Next ↴

Next ↴

Next ↴

After detecting devic: Next ↴

Note: It displays Configured device names

Next ↴

Next ↴

After Updating Device Configuration:

Next ↴

Storage unit (displays storage unit name)

Next ↴

Finish ↴

## Step 2: Configure the volumes.

Welcome Wizard: Next ↴

Click on Vnb ↴

Select device for vd. Config. TLD(0)

Next ↴

Start Robot Inventory

Next ↴

actually we have 30 volumes.

Next ↴

Select cleaning Cartridge

Select anyone of the Cartridge click [next] ↴

displays confirmation [next].

[Finish] ↴

## Step 3: Configure the Catalog Backup

[Next]

Welcome Wizard [Next]

Specify the Media Server Name

[Next] ↴

default locations

[usr|openv|netbackup|db]

[usr|openv|volmgr|database]

[usr|openv|var]

} for windows  
Servers only.  
? clients

Select Mediatype where you want to store Catalog backup info.

✓ Removable Media

Select the cartridge & click

✓ Alternative

Select the Media type

✓ Diskdrive

Specify the Patname | Catalog

When should a backup of the Catalog files occur?

✓ After each session of Scheduled, user (or) Manual

Next → finish

Step 4 : Create a Backup Policy

Note: for local system

Next → Next

Policy Name: MyPol

Policy type : Standard

Client list

Client name:

Vnb

Ok: Sol9  →

Backup Selectionlist

|sun

Backup Type :

- Full Backup
- Incremental Backup
- User Backup

Select frequency

Scheduled Job

- All day
- All day

→

↵

Next → Next → Finish

To see all policies.

Select

goto Console.

# mkdir Catalog

# mkdir sun

# cd sun

# touch Sun.txt Sam.txt fi fo fe ↵

Come to GUI 1001.

Right click Policiname → Manual backup

Select Schedules  ↴

double click on running Job id

To see go to Activity Monitor.

Complete Vnb is done on Error codes.

Goto Console

#&rm -r /Sun/

#Cd /sun/

Go to Netbackup tool

Restore the backup.

Select  
Backup, Archive & Restore.

Place the Cursor on Browse directory & Press Enter.

Select directory.

✓ Sun

↴

[Start restore] ↴

① goto Console.

② # cd /sun

③ # ls -l

## Performing Remote System Backup :

Taking backup of Solaris:10

Netbackup ~~client~~ client : Wstsun1H

# cat > l.ghosts

+

Ctrl+d ↴

# vi /etc/hosts

200.200.0.15 vnb

:wq! ↴

# mkdir /Sol-10

# cd /Sol-10

# Prod

# touch a b c d e f

# cd

# Ping vnb

## Come to Netbackup Master Server:

# cat > l.ghosts

+

Ctrl+d ↴

# vi /etc/hosts

200.200.0.14 westsun14

:wq!

# Ping westsun14

# ssh westsun14

:

goto Gui Console(TOO)

Select **Vnb(Master Server)**

Create a Backup Policy

↓  
**Next**

Policy Name Sol10 Pol

Type Standard

Add the client

**Add**

client name:

westsun14

H/W OS.

Sol.9

**OK** → **Next**

Backup Selection list

Add

{Sol.10}

Backup Type.

Next

Next

All day

All day

Add

Finish

Click on Policies.

Sol10Pol

↓  
Clients

Rightclick on

Client name

→ Install  
UNIX Client S/W.

Select the client hostname

Add >>

Install these

Install Client S/W

[Close]

Right click Policy Name

Select Manual Backup

To enable Client Side enable Postings

Hpcd → 13782

234

```
# cd /usr/openv/netbackup/bin  
# ./bpcd -Port 13782  
# rm -r /sol.10  
# cd /sol.10
```

## Performing backup of Aix Machine.

Ping  
# 200.200.0.156

```
# uname -a
```

```
# cat > /etc/hosts
```

Same as Sol.10 os.

Host name: vostro101

rx6500

## Uninstall

Yes | Pkgm VRTSnetbp

rm -r /opt/openv

rm -r /usr/openv

Yes | Pkgm VRTSnetbp

13/11/09

## SYSTEM MESSAGING

[etc|syslog.conf]

The Main Configuration file for all System & Log messages is

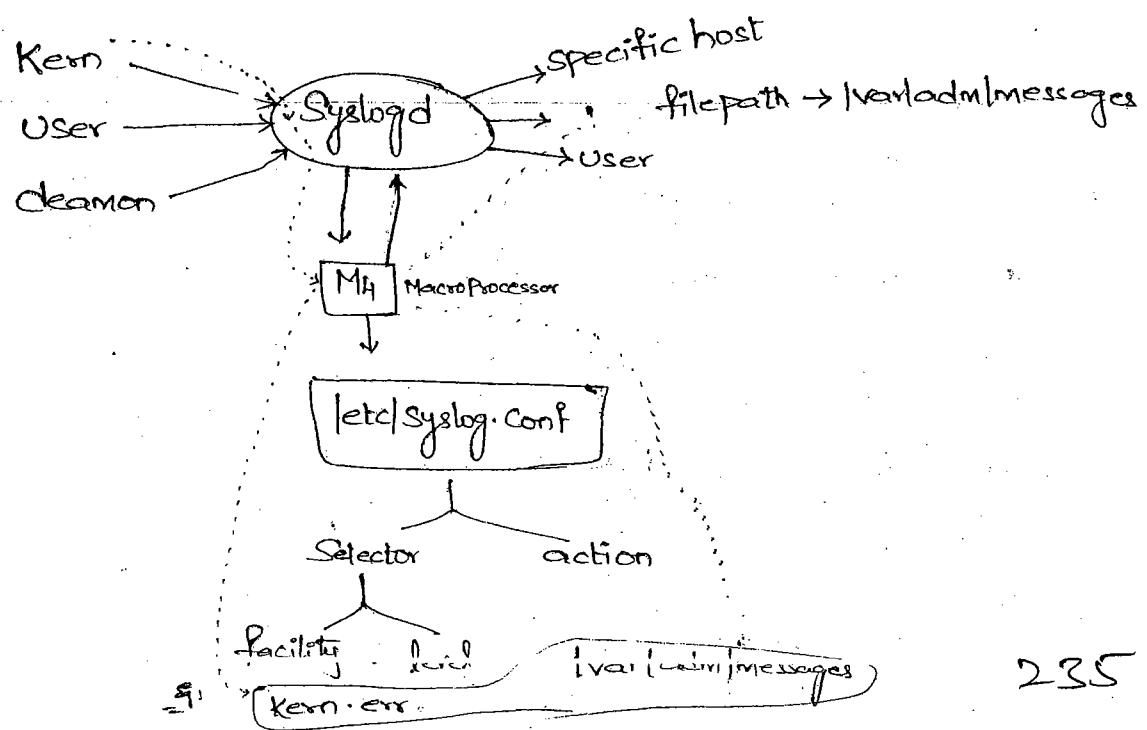
[etc|syslog.conf]

| Type of error | Where to send |
|---------------|---------------|
| *.*           | @loghost      |
| Kern.err      | ldev Console  |

Syslogd

daemon is responsible for reading the etc|syslog.conf  
- Sending the error msg to appropriate host

etc|syslog.conf & Syslogd daemon are responsible for facilitating the system messages.



/var|adm|Messages

- is the file responsible for maintaining each & every message generated.

It contains

#tail -f /var|adm|Messages

that error? pid? why error? resolve?

Syslogd daemon is responsible for catching the messages & it sends to appropriate hosts & specific file path.

M4 Macro: is a Micro Processor. It's a binary level program.

It acts as an filter & sends messages according to their criticality to the specific Path.

/etc|Syslog.conf:

It's a main config. file for system messages.

- This file logically divided into two parts.
  - \* Selector field
  - \* Action field

Under Selectorfield there is 2 locations

- \* facility & level
  - facility & level are separated with ':'.
  - facility represents what type of error it is
  - level represents criticality of the error.
  - Action represents from where we need to generate & send the messages.

## Types of Errors:

\* Emergency Messages

\* Debug

\* Notice

\* Warning

\* Info

\* Critical

\* Alert

etc.,

## Selector field & level options:

| <u>Level Selector</u> | <u>Priority</u> | <u>Description</u>                                                           |
|-----------------------|-----------------|------------------------------------------------------------------------------|
| Emerg                 | 0               | - Panic Conditions that are broadcast to all the users.                      |
| Crit                  | 1               | - Warning abt critical conditions such as hard disk errors.                  |
| err                   | 2               | - Error Messages.                                                            |
| debug                 | 3               | - while debugging it generate the msgs.                                      |
| Warn                  | 4               | - Warning Messages.                                                          |
| Notice                | 5               | - Non error Conditions                                                       |
| Info                  | 6               | - it displays complete info regarding the messages.                          |
| Alert                 | 7               | - Conditions that should be corrected immediately like a corrupted database. |

To direct the Messages on the Remote Machine

# vi /etc/hosts

200.200.0.1 westsun1

200.200.0.15 westsun15

} loghost

:wq!

# vi /etc/syslog.conf

\*.\* ifdef (LOGHOST @westsun15....)

↓ facility ↓ level

:wq!

#svcadm restart Sendmail

#logger -P

→ Priority.

↓ Generating Messages

#logger -P 0 "System is shutting down. Please ignore..."

## Security Issues:

If User Cannot login.  
This might be  
the reason.

{  
etc/default/login  
#Console.  
  
etc/nologin  
  
#Passwd -s  
[K-status.]

#vi etc/default/login

~~#RETRIES~~

#RETRIES = 5 → 3

↓  
uncomment  
it

:wq!

#vi etc/security/Policy.conf

Lock after Retries = No → Yes.

:wq!

To Monitor how Many Users are failed to login.

# touch /var/adm/loginlog

Eg: Login: u1 ↵

Password: xxx ↵  
wrong

Login incorrect

Login: u1 ↵

Password: xxx ↵  
wrong

Login incorrect

Login: u1 ↵

Password: xxx ↵  
wrong

Password is locked.

To see the status of user.  
Passwd

# Passwd -s u1 ↵

lk

# cat /var/adm/loginlog ↵

uname terminal time

To unlock the user.

# Passwd -u u1 ↵

## How to Monitor Super User Logs?

U1] \$ su - U2

Password

U2]\$

**[/var/admin/sulog]** : Responsible for maintain Super User Logs.

If this file is removed, it automatically creates a file while accessing the SU - Command.

**[/var/admin/lastlog]** : Before Rebooting who are the last logged in users in the System.

## Performance Tuning:-

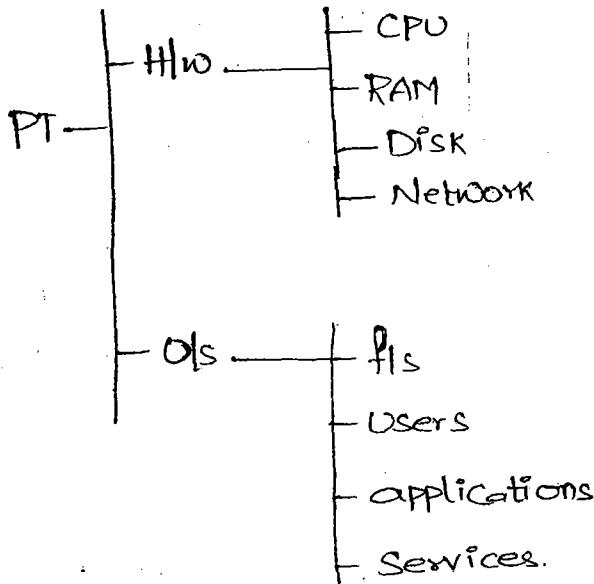
**# lockfs -a** : To lock a filesystem.

If a filesystem is locked then no user can login) create files. Even root user also.

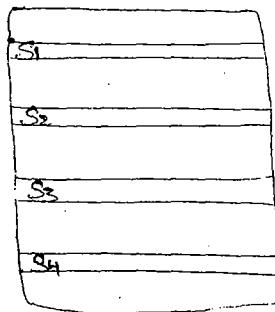
**# lockfs -w /Users**

**# lockfs -x /Users**

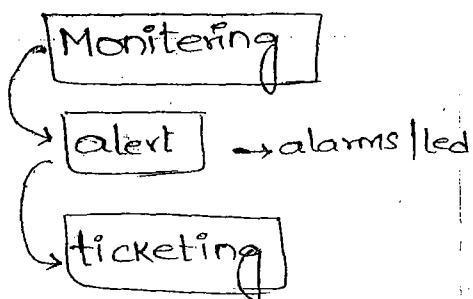
Performance: is depending upon | operating system, Application, Hard disk, CPU, RAM, Network etc,



- To Confirm whether the Server is running fine or not  
 regularly we need to monitor the above resources.



Rack Mountable Server



} These tools are used to know the exact problem. Which part has the problem. Whether HW level (or) O/S level anything it will tell.

## Monitory:

CPU: We have to Monitor Process, How Many Users are Exec, wheather Sleeping Process | Active Processes.

`#Ps -ef` : to display Processes.  
`#Ps -a`

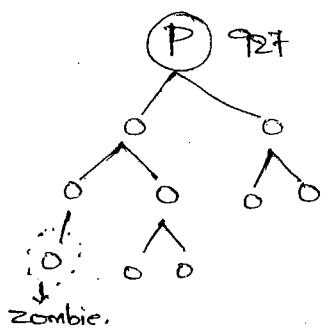
Note: While killing any Process the Suggestable way is Killing Process id's.

`#Pkill` is used to kill Process with Process name  
`#Kill` is used to kill Process with Processid.

`#Kill -9` : Sure shot killing

`#Kill -8` : Abrupt killing

`#Kill -9 927`



A Process which doesn't have Parent & Child Relation is called Zombie process | Orphan.

- Kernel will try to Kill the Zombie Process.

- As an Admin. we cannot kill Zombie Process.

`#kill -HUP <id>`

: without Stopping Process | Services it Restarts the Services.

`#Prstat`

: Responsible for displaying Process Statistics.

`#SdtProcess`

: Opens GUI tool regarding Process Info.

**#Pstat -a** : Including User names it lists. : list all the users accessing the Process.

**#Rstat -U <username>** : for Specific User.

### Pstat:

- It Shows dynamic updates of Process.
- It regularly Checks the Process Which are Consuming high CPU Resources & if Possible it Recycle the Unnecessary Processes.
- It Shows the Percentage of CPU Resource Utilized.
- It Even shows Priority of the Processes.

**#Ptree** Used to monitor the hierarchical process.

**#Ptree <pid>**

**#top** : like to Pstat. but for this cmd we need to install Pkg

\* If the Process Status is **<defunct>** then it is Zombie Process.

**#Sar** (System Activity Report)

**SUNWaccu** Pkg for Sar Cmd. | install full H/H Cd.

**#Sar 10 5**

**Sar <regular intervals> <how many times want to display>**

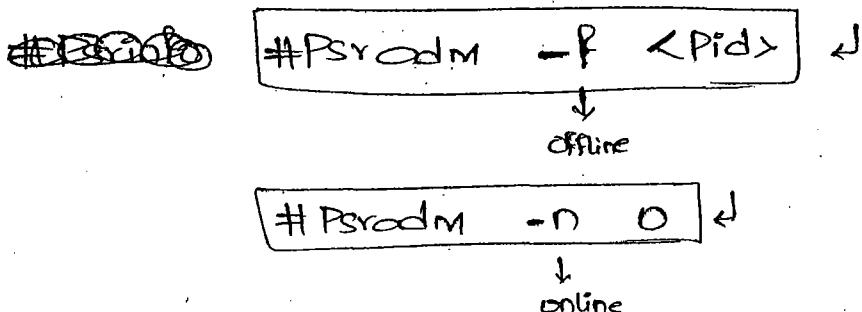
→ By executing Sar cmd. we can find out the CPU Idle time & Waiting I/O operations.

→ Waiting I/O operations are nothing but how much time the controller is taking to do Read/Write Operations.

**# Psrinfo** : Used to know how many Process we have in a system.

| <u>Id</u> | <u>P.states</u> |
|-----------|-----------------|
| 0         | online          |
| 1         | online.         |

To remove the Processor we need to make it Offline.



### 2 types of Process

Symmetric

Asymmetric : Share o/s resources & capping resources.

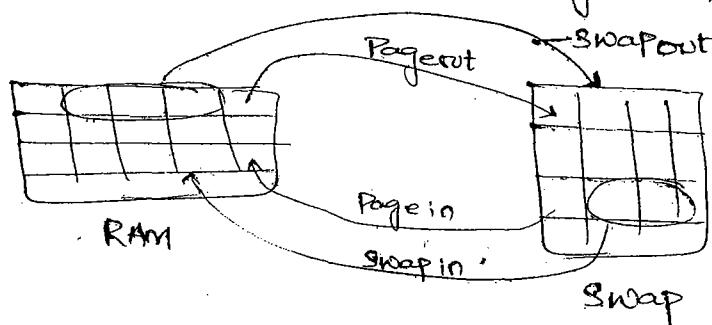
**# Psrinfo -v** ↓ : brief info regarding the Processor.

All Workstations will work on "Sparcq" Processor

**#Prtdiag** ↓ We can see Processor Speed.

### RAM:

**#Vmstat** ↓ Responsible for displaying the Virtual memory statistics. (Page ins Pageouts)



→ Sar Cmd & Vmstat Cmd's are C-Programmed. by executing Vmstat Cmd regularly we need to monitor Pagein, Pageout & Swapin, Swapout.

**#Pagesize**: is a cmd used to display current pagesize.

<sup>default</sup>  
8kb — Sparc

4kb — x86

To See Ram Size

In obp level      **#banner**

In Os level      **#PrtConf -v | grep -i Mem**

**#Swap -l** : used to display swap area.

**#ipcs** : used to monitor the CPU resources as well as RAM resources.

### \* Disk:

**#iostat -en** : to know the disk is having S/w errors | H/w errors  
If S/w errors (or) H/w errors are more than 15

**#iostat -En** : displays brief information. (Vendor ...)

**#iostat -cmdx** : responsible for monitoring R/W operations on a disk

**#cfgadm -al** : status of the disk (Configured | not)

Network Monitoring: Whenever an Interface is initialised by Kernel, a message is being updated under /var/adm/messages file

- The Configured Interface may be full duplex / half duplex.

How can I know whether the Interface is full duplex or half duplex.

# ifconfig -get DevName link-Speed

0 → half duplex

1 → full duplex

/etc/Path\_to\_inst

: Maintain all Configured linc cards.

# ping -v | grep net : How many N/w interface cards are configured.

# netstat : Responsible for displaying N/w Statistics.

# netstat -a : Used to display all the Port information which are idle / listen / available / nonavailable Ports

# netstat -r : gives info. Regarding gateways.

# Ping -s <R.IPs> : Monitoring N/w Packets.

\* Kernel :- All Kernel tunable Parameters are located  
under **/etc/system**

**# isainfo -b** ↳ : displays bit rate (32 bit) / 64 bit)

**lockfs -a ↳**

→ locking

**lockfs -w [users]**

**# lock -U [users]**

→ unlocking

**# fuser -CUK [users]**

**# umount [users]**

} If at all the umount not successful then there may be chance of the user accessing the M.P. so we can execute this cmd.

**# Sysdef** ↳ : System Wide info. is displayed.

### \* Services :-

- If the System is in Maintenance Mode, then follow these

**# cat /etc/vfstab** : Check improper entries.

**# share**

**# cat /etc/auto-Master** } for nis  
**# cat /etc/auto-home** }

**# format** ↳ | check any cylinders are overlapping (or) not

**# svcs -a | grep file** : Should be online.

If f/s status is in maintenance

**# sradm clear <fsid>**

if in offline

- \* Svcadm disable <fmri>
- # Svcadm enable -r <fmri>
- \* Svcs -d <fmri> : Child level Services
- # Svcs -D <fmri> : Parent level Service.

if in maintenance

- \* Svcs -a | grep file
- # Svcs -a | grep mfile
- # Svcs -a | grep inetd
- # Svcs -a | grep network/physical
- # ifconfig hme0 unplumb.  
Plumb

assign IP

still doesn't come then

- # cd /lib/svc/method
- # .|Fs-root
- # .|Fs-user
- # .|Fs-minimal
- # .|Fs-local

# Showver

: how Many Patches are installed

Patchadd -P

### \* Kernel

# vi /etc/system

moddir:

# Modload : Command used to load drivers. invoked by Kernel

# Modinfo : Available Modules.

# Modinfo -c : Status (Installed/not)

# Modinfo -i <id> : To see Particular module Info.

# Modunload -i <id> : To unload a Particular driver.

### root device & root -f ls Config

# rootfs

# rootdev

### Exclude

Modules appearing in the moddir Path which are not to be loaded.

exclude: Pool

ForceLoad: If any module/driver doesn't load mention here it will forcibly loads.

### Set:

Set an integer variable in the Kernel (or) a module to a new value.

# Prstat -c

: to see Current & Previous Status of Process

# Prstat -p

: to see no of Processes.

#dmesg : How many messages. & Provides resolution.

OK> Show-nets : net cards

OK> Show-devs : devices

OK> Show-tapes : tape drives

OK> .Speed

OK> .Version

OK> Words : shows all Cmds used at: Obj.

OK> help diag

OK> Ctrl+P : to see Previous Cmd.

OK> Ctrl+N : next cmd.

OK> Ctrl+B : to modify

14/11/09

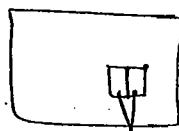
## Hard Disk Technologies:-

IDE → Transfer rate 40MBps

MB  
→ Bytes  
Mb  
→ bits

SATA → 350Mbps

Dual Core



Bus speed - FSB

1333

Core 2 duo

Bus speed - FSB  
1666

VMware  
Virtual bo  
Citrica

Logically divided into 2

## Scsi:-

Narrow Scsi → 0 → 7      ⑦ initiator

Wide      } 0 - 15      ⑦ initiator

ultrawide      } 0 - 15      ⑦ initiator

Narrow scsi

50 pins.

Wide scsi

68 pins.

- Narrow Scsi supports 120 MBps.

- Wide Scsi supports 160 MBps.

- Ultrawide Scsi supports 320 MBps.

- hot pluggable
- hot swappable.

- to change the harddisk in hot pluggable first we need to take it into

Unconfigure state.

- We can directly change online in hot swappable.

Scsi

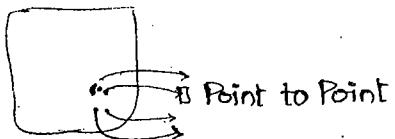


SAS

→ Speed 6Gbps.

Serial Attach Scsi

Latest & now using in Env.



## FC : (Fibre channel)

- Used in SAN (Storage Area Network).

Speed: 8Gbps & 4Gbps

↳ Siemens Tech. (Supports 1,2 Gbps)  
 ↳ Lucent (Supports 4,8 Gbps)

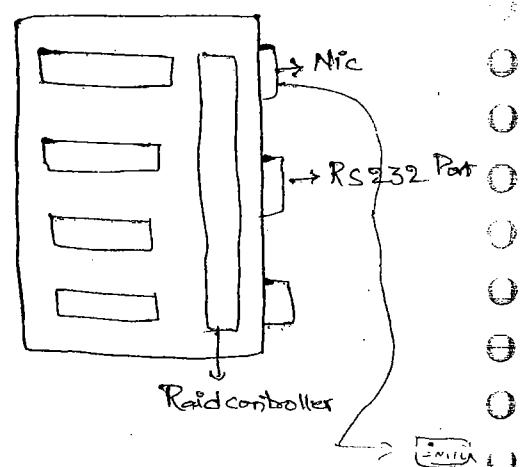
| <u>Scsi</u> | <u>FC</u> |
|-------------|-----------|
| Cat5e       | Cat5e     |
| de          | de        |
| ds          | ds        |
| dh          | dh        |

SVM → Raid Technologies are Purely implemented on Storage.

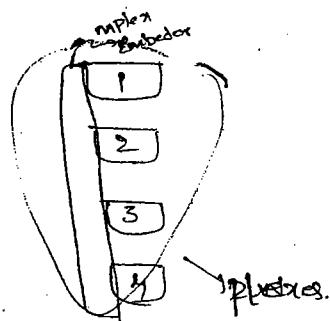
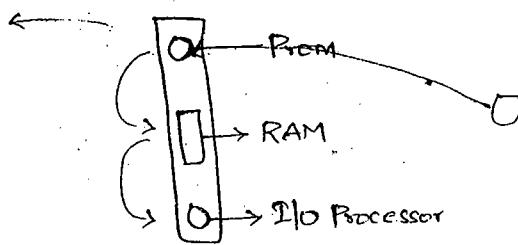
Storage has Redundancy Copy also.

## NAS: (Network Attach Storage)

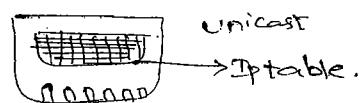
- Purely depend on NFS Technology
- NFS uses TCP/IP Protocol.



## Raid Controller



## Switch



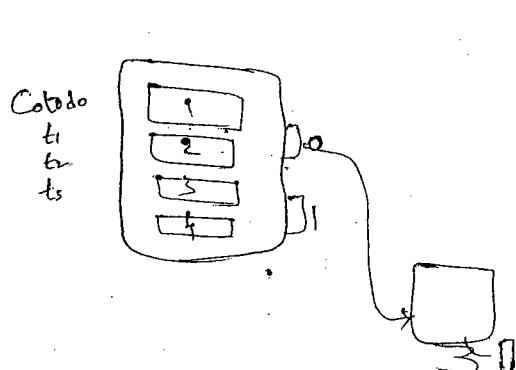
Non Manageable

Manageable. (Support 1 Gbps)



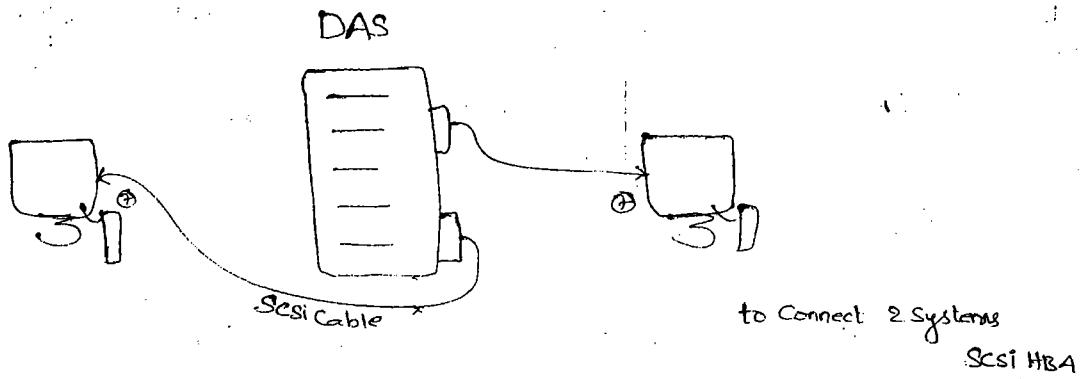
## DAS : (Direct Attach Storage)

### Assembled



### Brand Storage

Scsi HBA (Host Bus Adapter)



OK>Probe-Scsi ↳ : Internal disks

OK> Probe -scsi -all ↵ : External disks

Ok> Setenv : initiator\_id 6

Set value  
Default value

`eff eeprom ; grep boot -devices.` : displays boot disk.info.

# PrtConf -VP

