

# 第一章 概述

# 本章目的

- 介绍AIX操作系统的主要组成部分
- 描述UNIX和AIX的发展和变化

# UNIX概念

- 多任务、多用户的操作系统
- 功能丰富的可扩展、开放的计算环境
- UNIX同时又指一个功能强大的多任务、多用户的操作系统

# UNIX历史

- 60年代中期，AT&T贝尔实验室组织开发了一个叫Multics的操作系统。Multics的主要设计目标是支持大规模的计算工具，后来参加Multics开发的许多贝尔实验室的人员都参加了UNIX的开发工作
- 1969年贝尔实验室从Multics的计划中撤出，由于科学计算研究中心（Computing Science Research Center）的成员没有了计算环境，Ken Thompson、Dennis Ritchie和其他一些研究人员开发了一个基本的文件系统，该系统后来演化成了UNIX文件系统
- 在1986年，IBM发布AIX操作系统，最初是安装在6150微计算系统（Micro Computer System）以及后来的Personal Sys/2 和 System/370. 1990年，AIX 3安装在IBM发布的RS/6000上

## Bell Labs

60年代中期

1979 Version 7

1983 System V

1985 System v.2

1986 System v.3

1989 System v.4

## IBM

1986 AIX Version 1

1987 AIX Version 2

1990 AIX Version 3.1

1992 AIX Version 3.2

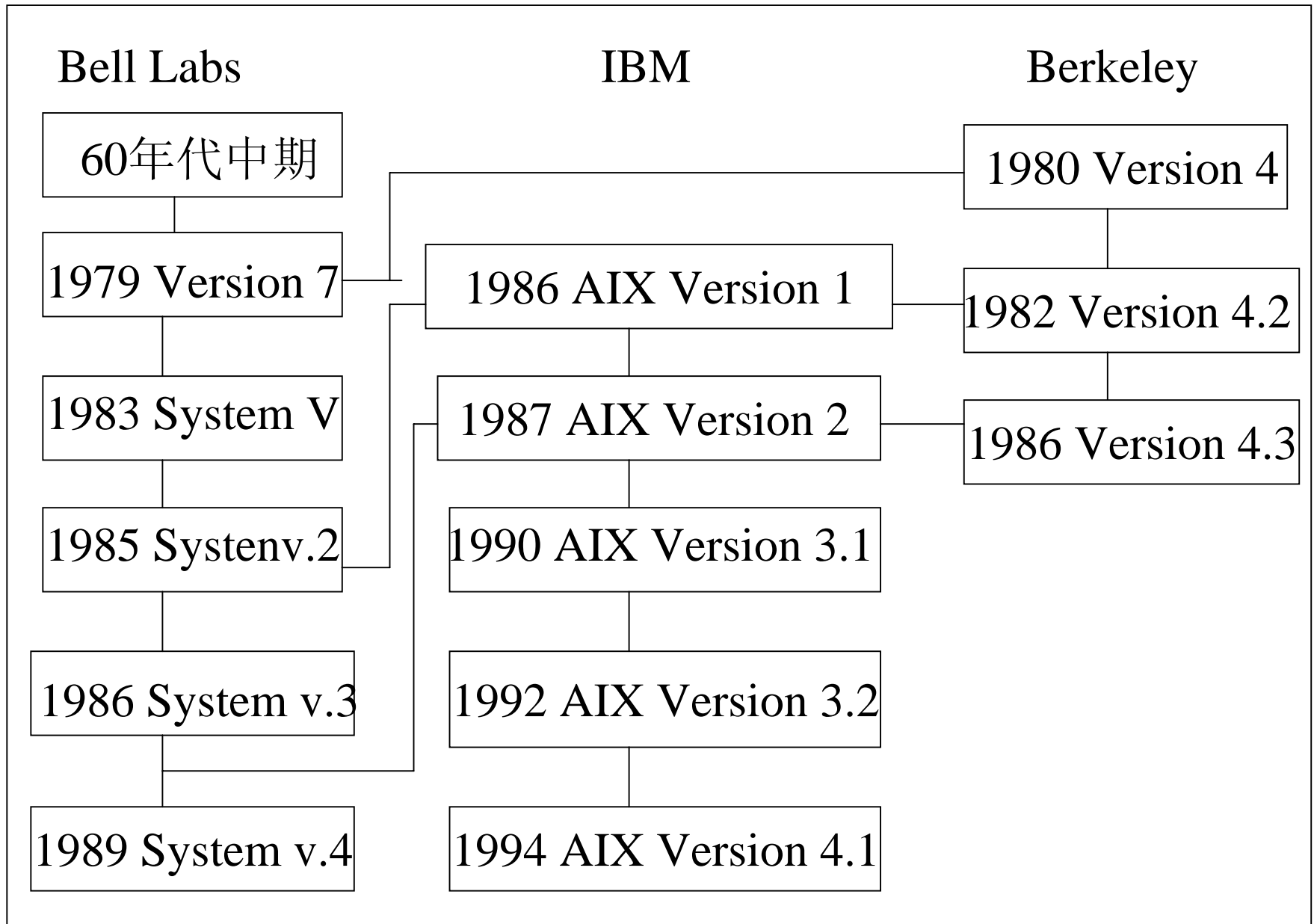
1994 AIX Version 4.1

## Berkeley

1980 Version 4

1982 Version 4.2

1986 Version 4.3



# AIX (Advanced Interactive EXecutive)

- Advanced Interactive Executive(AIX)是IBM基于UNIX基础之上发布的操作系统

# Open System 开放式操作系统

- 充分利用投资
- 满足客户需求
- 标准战略对策
- 扩展能力强
- 商业灵活性

# AIX 的主要组件

- 核心
- 文件系统
- shell
- 通讯/网络
- 文本
- 程序



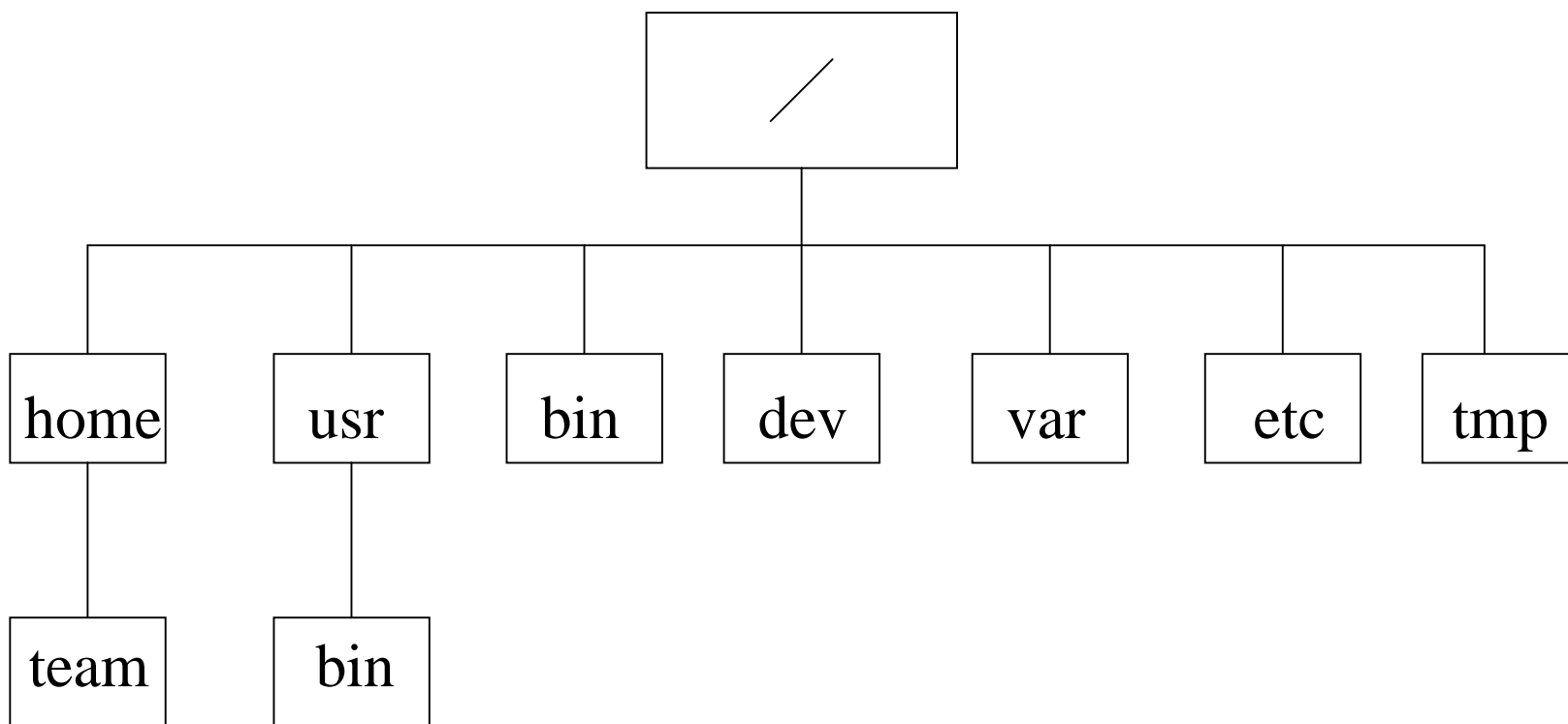
# AIX 的主要组件（续）

- 系统管理
- 在线文档
- Xwindows
- 其它工具
- 应用

# 核 心

- 多任务处理/多用户
- I/O设备的接口
- 管理磁盘上文件
- C语言环境

# 文件系统



# shell

## UNIX和用户的界面

- 几个有效的Shell
  - Korn
  - Bourne
  - C
- 缺省的Shell ---- Korn
- 命令解释器
- 可设计
- 支持处理的执行

# 系统管理

- 安装软件
- 定义用户帐号
- 设置外围设备
- 分配磁盘的存储
- 备份数据
- 监视性能
- 解决系统故障

# 第二章 系统的运行

# 本章目的

- 登录及退出系统
- AIX的命令结构
- 学会使用一定数量的命令

# 登录及退出系统

Login

Login: team01

term01's Password: \*\*\*\*

\$

Logout

\$ <ctrl-d> or

\$ exit or

\$ logout



# 密 码

## 创建或者改变密码

```
$ passwd
```

```
Changing password for team01
```

```
team01's old password:
```

```
team01's new password:
```

```
Enter the new password again:
```

```
$
```

# 命令的格式

\$ command options arguments

For Example :

\$ mail -f newmail

# 命令格式举例

RIGHT

WRONG

## 1 Separation

```
$ mail -f newmail
```

```
$ mail - f newmail
```

```
$ who -u
```

```
$ who - u
```

## 2 Order

```
$ mail -f newmail
```

```
$ mail newmail -f
```

## 3 multiple

```
$ who -m -u
```

```
$ who -m-u
```

```
$ who -mu
```

```
$ who -m u
```

# Date 和 cal 命令

```
# date
```

```
Thu Jan 27 05:34:40 TAIST 2000
```

```
#
```

```
# cal 2 1995
```

```
February 1995
```

```
Sun Mon Tue Wed Thu Fri Sat
```

```
1 2 3 4
```

```
5 6 7 8 9 10 11
```

```
12 13 14 15 16 17 18
```

```
19 20 21 22 23 24 25
```

```
26 27 28
```

# Who 和 finger 命令

```
# who
```

```
notes      lft0      Jan 23 09:18
notes      pts/0     Jan 24 01:19      (:0.0)
root       pts/1     Jan 27 05:34      (128.0.0.71)
```

```
# who am i
```

```
root       pts/1     Jan 27 05:34      (128.0.0.71)
```

```
# finger root
```

```
Login name: root
```

```
Directory: /                               Shell: /bin/ksh
```

```
On since Jan 27 05:34:17 on pts/1, 14 seconds Idle Time
    from 128.0.0.71
```

```
No Plan.
```

```
#
```

# 发送电子邮件 sending mail

```
$ mail team01
```

```
Subject:Meeting
```

```
hello everyone
```

```
<ctrl-d>
```

```
Cc:<Enter>
```

```
$
```

# 接收电子邮件 Receiving mail

YOU HAVE NEW MAIL

\$ mail

Mail [5.2UCB] {AIX4.1} type ? For help

“/var/spool/mail/team01”:2 messages 1 new

U 1 team05 Tues Jan 3 10:50 10/267 “Hello!”

>N 2 team02 Wed Jan 4 11:25 16/311 “Meeting”

?t<ENTER>

# Clear、echo and banner 命令

```
$ clear
```

```
$
```

```
$ echo Lunch is at 12.00
```

```
Lunch is at 12.00
```

```
$
```

```
$ banner Hello
```



# WC 命令

```
$ wc [-c] [-l] [-w] filename
```

选项

-c 计算字节数

-l 计算行

-w 计算字

```
$ wc myfile
```

```
17   126      1085 myfile
```

lines	words	总和
-------	-------	----

# 键盘的快捷方式

- `<ctrl-c>` 停止命令
- `<ctrl-d>` 结束传输或者文件
- `<ctrl-s>` 临时停止输出
- `<ctrl-q>` 恢复输出
- `<ctrl-u>` 擦除整行
- `<backspace>` 纠正错误

# 第三章 文件和目录

# 本章目的

- 描述不同的文件格式
- 描述AIX文件系统的结构
- 文件的绝对路径和相对路径
- 创建、删除和列出路径
- 复制、显示、打印、移动、删除和连接文件

# 文件的定义

- 数据的集合
- 字符或字节流
- 没有强加的结构

# 文件的格式

- 普通
- 路径
- 特殊文件    `/dev/tty1`

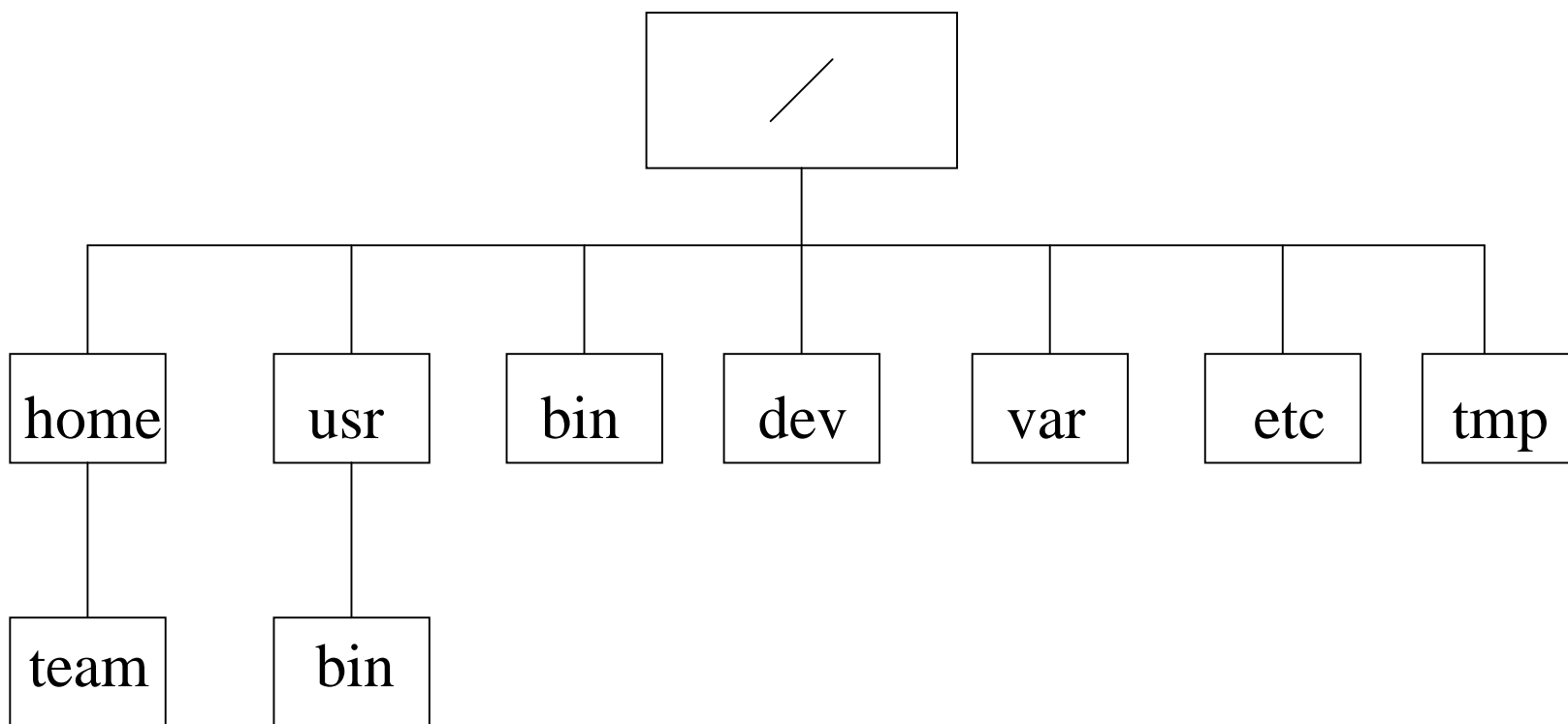
# Inodes

- 一个文件包含两部分
  - inode -- 文件的描述
  - 数据

## Inode 的 内容

- 文件的尺寸
- 权限
- 创建的日期和时间
- 最后被改动的日期和时间
- 连接数

# 文件系统的层次结构





# 路径的名称

- 类型
- 绝对路径
- 相对路径

Where am I

```
$ pwd
```

```
/home/team01
```

# 选择正确的路径

\$ cd doc

\$ cd /home/team01/doc

\$ cd

\$ cd ..

## 创建路径

```
$ mkdir dir_name
```

```
$ mkdir /home/team01
```

## 删除路径

```
$ rmdir dir_name
```

# 路径列表

```
$ ls dir_name
```

```
$ ls -a
```

```
$ ls -R
```

# AIX文件名称

- 包含 大写键、小写键、数字、#、@、\_
- 不包含空格
- 不包含以下字符  
\* ? > < / ; \$ \ ‘ “
- 不能以 “+” 或者 “-” 开头
- 最长文件名 255

# 复制文件

```
$ cp source target
```

```
$ cp file1 file2 ..... target_dir
```

# 移动/重命名文件

```
$ mv source target
```

例如:

```
$ mv t.letter ../doc/letter
```

```
$ mv ../doc/mon_report
```

# 连接文件

```
$ ln source_file target_file
```

# 显示文件内容

```
$ cat filename1
```

```
$ pg filename1
```

```
$ more filename1
```



# 删除文件

```
$ rm file1 file2 ...
```

```
$ rm -I file1
```

# 打印文件

```
$ qprt filename filename1
```

```
$ qchk
```

```
$ qcan -x number
```

# 第三章 文件的权限

# 列出文件

```
$ ls -l
```

## touch 命令

命令语法: touch filename

```
$ touch suba newfile
```

# 文件的保护和权限

rwx

rwx

rwx

user

group

others

一个普通文件

r = 可以查看文件内容

w = 可以修改文件内容

x = 可以执行文件

# 一个路径

r : 可以查看文件夹下的文件

w : 可以在文件夹下创建和删除文件

x : 可以进入文件夹或者访问文件夹下的文件

# 改变文件的权限

```
$ chmod mode filename
```

u = owner of the file

g = owner's group

o = other users on the system

+ = add permissions

- = remove permissions

= = clears permissions and sets to mode specified

```
$ chmod go+w newfile
```

	<b>user</b>	<b>group</b>	<b>others</b>
符号	<b>rwX</b>	<b>rw-</b>	<b>r--</b>
二进制	111	110	100
	4+2+1	4+2+0	4+0+0
八进制	7	6	4

```
$ chmod 664 newfile
```

# 缺省的文件权限

file	-rw-r--r--	644
------	------------	-----

directory	drwxr-xr-x	755
-----------	------------	-----

umask

/etc/security/user umask 022



# 第三章 shell基础

# 本章目的

- 通配符
- 输入输出重定向
- 管道
- 命令组

# The Shell

- Korn(ksh) or Bourne (bsh) or C (csh)

## 用户和AIX的接口

- 命令解释器
- 多任务
- 包含内容丰富的程序语言

# 文件名中的字符代用字

- 单字符代用字

?

- 多字符代用字

\*

- 包含代用字

[ ] ! [ - ]

# 标准文件

- 标准输入（0）
- 标准输出（1）
- 标准错误输出（2）

# 输入输出重定向

- 输入重定向
  - `command < filename`
- 输出重定向
  - `command > filename`
- 错误重定向
  - `command 2> filename`

# 管道及过滤

- 管道
  - `cmd1 | cmd2`
- 过滤
  - `command | filter`

# 分向输出

- tee

# 命令集合

- `command1 ; command2 ; .....`



# 续行

● .....  
.....

.....

／

● .....  
.....

# Shell 变量

## 定义个人设置

- HOME
- TERM
- PATH

## 列出变量的设置

`$ set`

## 删除变量

`$ unset`

# 第八章 vi (选)

# 本章目的

- 建立和编辑文件
- vi

\$ vi myfile1

## 两种操作模式

\* 命令状态

\* 文本状态

~

~

~

~

~

“vifile”[New file]

# 单字符命令

- 插入:

a

A

i

I

<esc>

<up\_arrow>

k

<left\_arrow> h

l

<right\_arrow>

j

<down\_arrow>



# 删除

x

dd

dw

d\$

d0

:20, 40d

退出

:q

:w

:x

:wq

# 第九章 进程

# 本章目的

- 创建及调用scripts
- 定义进程
- 父进程和子进程

# 进程概念

- 任务
- 运行的程序
- PID

```
$ echo $$
```

# Login 进程环境

- Login

program	-ksh
uid	john
gid	staff
files	/dev/tty1
PID	201

# 父进程和子进程

PID	PPID
201	1
206	201
207	206

```
$ echo $$
```

```
201
```

```
$ ksh 建子SHELL
```

```
$ echo $$
```

```
206
```

```
$ date
```

```
tue sept 5 11:18:26 gmt 1995
```

```
$ <ctrl-d> 退出子SHELL
```

```
$ echo $$
```

```
201
```



# Shell 变量

- 输出变量export
- \$x=4
- \$echo \$\$
- 4687
- \$export x

输出shell变量export

```
$ x=4
```

```
$ echo $$
```

```
4687
```

```
$ export x
```

```
$ ksh
```

```
$ echo $$
```

```
4731
```

```
$ echo $x
```

```
4
```

```
$ x=400
```

```
$ <ctrl--d>
```

```
$ echo $$
```

```
4687
```

```
$ echo $x
```

```
4
```

# Shell Scripts

- 创建script
- \$cat script1
- date
- pwd
- ls -l

# 使用脚本

```
$ chmod +x script1
```

```
$ ./script1
```

# 返回码

```
$ ?
```

```
$ date
```

```
$ echo $?
```

```
0
```

# 第十章 进程的控制

# 本章目的

- 进程监视
- 调用后台进程
- 中断进程
- 显示有用信息
- nohup
- 控制 jobs
- 定义系统进程

# 进程监视

- `$ ps -f`
- `UID PID PPID ...TTY ...COMMAND`
- `john 201 1 ...1 ...-ksh`
- `john 206 201 ...1 ...ksh`
- `john 209 206 ...1 ...ls-l`

# 中断进程

- 前台进程
  - ctrl-c
  - kill
- 后台进程
  - kill



```
$ ps -f
```

UID	PID	PPID	...TTY	...COMMAND
john	206	201	...1	...ksh
john	209	206	...1	...ls -R

```
$ kill 209
```

```
$
```

# 运行 Long Processes

## **nohup**

```
$ nohup ls -R / > out &
```

```
$ nohup ls -R / &
```

# Daemon

- 守护进程

# 第十一章

## 客户化用户环境

# 本章目的

- 了解profile用途
- 改变PATH和PS1变量

**\$ cat .profile**

```
PATH=/bin:/usr/bin:/etc:$HOME/bin:
```

```
PS1='$PWD $'
```

```
export PATH PS1 ENV
```

```
ENV=''$HOME/. kshrc''
```

# PATH

PATH=/bin:/usr/bin:/etc:\$HOME/bin:.

# PS1

PS1='\$PWD \$'

ENV=''\$HOME/.kshrc''

# Aliases

```
$ alias p='ps -ef'
```

```
$ alias l='ls -l'
```

```
$ alias
```

```
history=fc -l
```

```
l=ls -l
```

```
p=ps -ef
```

```
r=fc -e
```



```
$ 1
```

```
-rw-r--r-  1  joe  staff   524  Sep 19  11:31  fleas
-rw-r--r-  1  joe  staff  1455  Jan  23  11:18  walrus
```

```
$ unalias 1
```

```
$ 1
```

```
1:command not found
```

```
$
```

# History

```
$ history
```

```
2 cd /u/payroll
```

```
3 ls -l
```

```
4 mail
```

```
5 history
```

```
$ r m
```

```
No mail for class
```

```
-or-
```

```
$ r 4
```

```
No mail for class
```

# 第十二章 AIX工具

- find
- grep
- head & tail
- cut
- sort
- dosread & doswrite & dosformat

# 查找Find

命令格式: `find path expression`

```
$ find . -name sum -print
```

```
./color/sum
```

```
./shape/sum
```

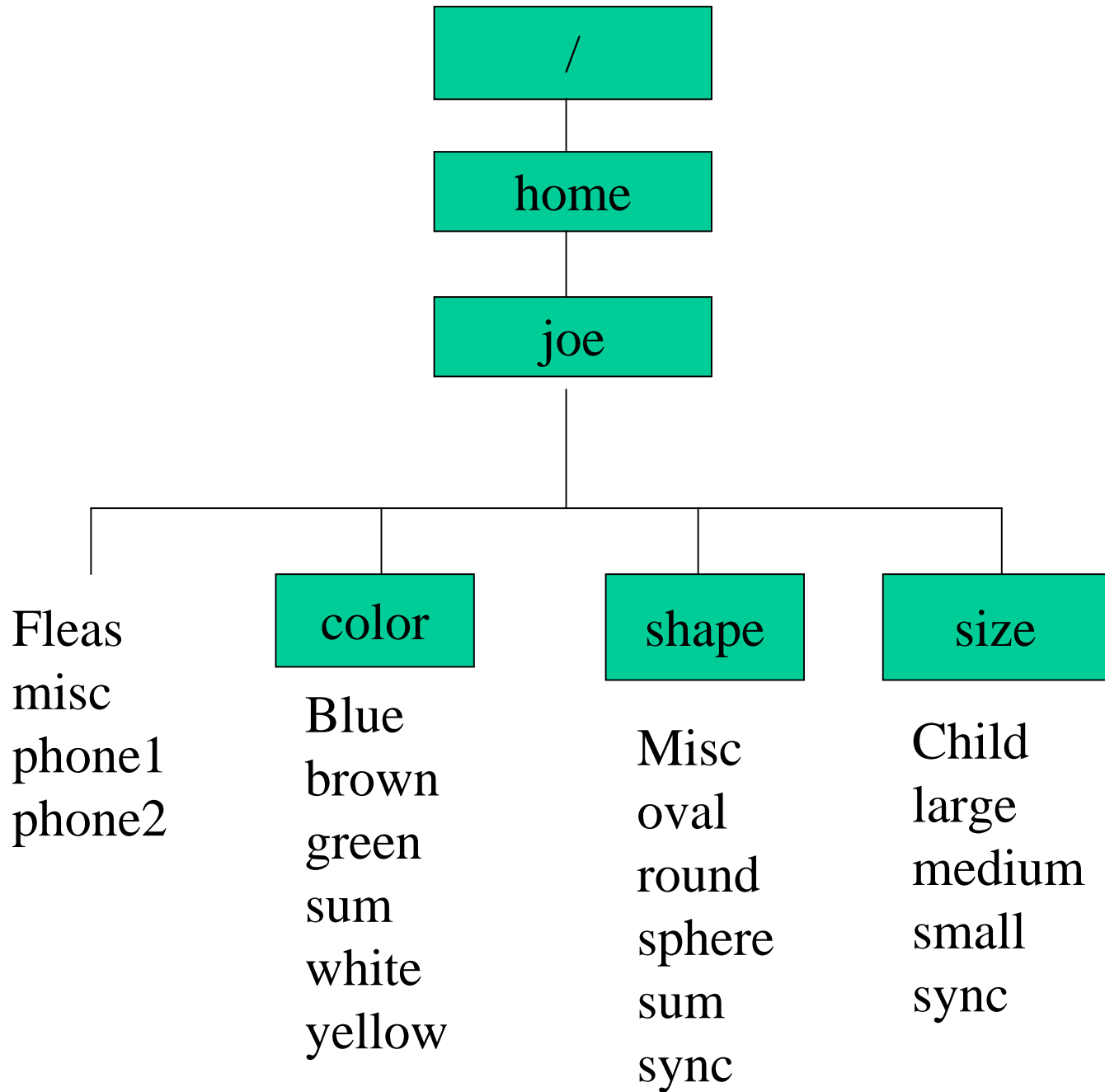
```
$ -
```

# 选项Execut

```
$ find .-name 'm*' -exec ls -l {} \;
```

```
-rw-r--r-- 1 joe staff 83 Jan 11 15:55 ./shape/misc  
-rw-r--r-- 1 joe staff 21 Jan 11 16:01 ./size/medium  
-rw-r--r-- 1 joe staff 38 Jan 11 15:34 ./misc
```

```
$ -
```



# 选项ok

```
$ find . -name m\* -ok rm {} \;
```

```
< rm ... ./shape/misc>? Y
```

```
< rm ... ./size/medium>? Y
```

```
< rm ... ./misc >? N
```

```
$ -
```



# 其它选项

- type f  
d
- size +n  
-n  
n
- mtime +x  
-x
- user user

# Find Examples:

```
$ find . -name 's*' -type f -size +2 -exec ls -l {} \;  
-rwxr-xr-x 1 joe staff 1512 Jan 11 15:43 ./color/sum  
-rwxr-xr-x 1 joe staff 2148 Jan 11 15:57 ./shape/sum
```

```
$ find . -perm 644 -mtime +4 -print  
./shape/misc
```

```
$ find . \( -name fleas -o -name misc \) -print  
./misc  
./shape/misc  
./fleas
```

```
$ find / -name 'security' -print  
/usr/lpp/bos/inst-root/etc/security  
/usr/lib/security  
/etc/security
```

```
grep
```

```
$ grep 68588168 phone
```

# cut

-c

-f

-d

```
$ cat /etc/passwd
```

```
root:!:0:0:The Super User:/:/bin/ksh
```

```
daemon:!:1:1::/etc:
```

```
bin:!:2:2::/bin:
```

```
sys:!:3:3::/usr/sys:
```

```
adm:!:4:4::/usr/adm:
```

```
$ cut -f1,3 -d: /etc/passwd
```

```
root:0
```

```
daemon:1
```

```
bin:2
```

```
sys:3
```

```
adm:4
```

**sort**

-d

-r

-n

\$ cat animals

dog.2

cat.4

elephant.10

rabbit.7

\$ sort animals

cat .4

dog.2

elephant.10

rabbit.7

# head

```
$ head -5 myfile
```

```
$ ls | head -12
```

# tail

```
$ tail -200 file
```

```
$ tail +200 file
```

# 传输DOS数据文件

\$ dosdir

\$ dosread file 1 .doc file1

\$ doswrite file 1 file 1 .doc

\$ dosformat