

```
1. sdimage: An EXT2 file system /
    |---bin : commands
    |---dev : specail files
    |---etc : passwd
    |---user: user home dirs
```

2. Wanix Kernel:

	Process	I/O buffer	Device
	Management	EXT2 FS	Drivers

```
3. Wanix Start:
  ts.s  : initialize SVC, IRQ, and other modes stacks;
         copy vector table to PA 0;
         make page table for P0, enable MMU for VA->PA mapping
         call main() in t.c

  main(): initialize IRQ handlers, I/O devices using interrupts;
         kernel_init: initialize kernel data structures;
                     create and run P0;
         mount_root:  mount the root file system on sdimage

         P0 creates P1 by kfork("init"): load init as the Umode image of P1
         P0 switch to run P1 in Umode

  P1: run init in Umode:
     fork() P2, P3, P4 to run login on different terminals;
     P1 waits for ZOMBIE child;

  P2: run login /dev/tty0
  P3: run login /dev/ttyS0
  etc.
```

=====

```
P2: login on /dev/tty0  (console=LCD, keyboard)
   open OWN terminal as 0, 1, 2
   login:
   password:
   After user login, becomes the user's process;
   exec to sh.
```

```
sh: loop:
   Ask for user input line: cmd arg1 arg2 ... argn

   Simple commands: fork() a child to execute cmd;
                   wait for child termination;
                   repeat loop
```

=====