

## Initrd (イニシャルRAMディスク)

組み込みLinux開発をしている時や、PCにLinuxをインストールする時に、よく聞くフレーズとしてInitrd(イニシャルRAMディスク)というものがあります。Initrdとは一体なんなのでしょう？

### Initrdとは？

Initrdとは、その名が示す通り、イニシャル(一番最初に、つまりブート時に)使われるRAMディスクです。イニシャルという言葉は、「ブート時」ということは表すので、まだわかりやすいのですが、RAMディスクとは一体なんなのでしょう？

### RAMディスクとは？

RAMディスクとは、RAM(メモリ)上に作成したディスク(二次記憶装置。例えばハードディスクなど)のことなのです。もっと分かりやすく言うと、メモリ上に仮想的に作成したハードディスクのことです。その仮想的なハードディスクは、ユーザからは本当のハードディスクにしか見えません。RAMディスクを使用すると、ハードディスクがつながっていないくても、ユーザからはハードディスクがあるように見えるのです。

### Initrdは何の役に立つの？

話をもとに戻しましょう。ではInitrd、つまりブート時にメモリ上に作成した仮想的なハードディスクは何の役に立つのでしょうか？

実はInitrdを使う理由は、ハードディスクがないシステムで、ハードディスク(つまり読み書き、そして保存できるデバイス)を使いたいからです。

組み込みシステムでは、一般的にハードディスクは使われません(携帯電話を想像してみてください)。そのようなシステムにおいて、あるファイルの編集(読み書き)そして、保存をしたいときは、どのようにすれば良いのでしょうか？その時に、登場するのがイニシャルRAMディスクなのです。組み込みシステムにFlash ROMがある場合には、JFFS2などのファイルシステムを使って、ファイルの読み書き保存ができるので、Initrdを使う必要はないのですが、実は、あえてイニシャルRAMディスクを使う場合があるのです。Flash ROMには、消去限界というものがあり、Flash ROM上でファイルを編集していると、少しずつではありますが、Flash ROMの寿命が減って行ってしまいます。そこで、ファイルの編集は、イニシャルRAMディスク上でを行い、ファイルが完成して保存する段階になってはじめて、Flash ROMを使うことにより、Flash ROMの寿命をできるだけ延ばそうと工夫する場合があります。

### InitrdはPCでも使われる！

PCには、ハードディスクが普通は搭載されています。それにも関わらず、Initrdが使われています。なぜハードディスクが存在するのにも関わらず、Initrdが使われているのでしょうか？

それは、Linuxがブートする仕組みに起因しています。メモリ上にLinuxがロードされた後、Linuxがしなければならない仕事の一つとして、ルートファイルシステムをマウントするということがあります。

もし、ルートファイルシステムが特殊なデバイスの上に存在した場合、Linuxカーネルがその特殊なデバイス用のデバイスドライバを持っていない可能性が十分にあり得ます。そのようなとき、Initrdを使って一時的にメモリ上にルートファイルシステムを展開し、それをマウントした上で、その一時的なルートファイルシステムから、特殊なデバイス用のデバイスドライバを読み込んで、改めて特殊なデバイスの中にある本当のルートファイルシステムをマウントするという二

段階のマウントをします。

もちろん、Initrdには予め、その特殊なデバイス用のデバイスドライバを入れておかなければなりません。

その特殊なデバイス用のデバイスドライバをあらかじめLinuxカーネルに組み込んでおけば、このような手間をする必要はない(Initrdを使う必要はない)のですが、ディストリビューションに含まれている万人向けのLinuxカーネルにはデバイスドライバはほとんど含まれておらず、ほとんどのデバイスドライバをモジュール化していますので、Initrdを使う場合が多いという訳なのです。

それでは、ちょっと私のPC用のInitrdをみてみましょう。Initrdは普通は/bootにあります。

```
# cd /boot
# ls
System.map-2.4.27-2-686  config-2.4.27-3-686      initrd.img-2.4.27-3-686
System.map-2.4.27-3-686  grub                      vmlinuz-2.4.27-2-686
config-2.4.27-2-686      initrd.img-2.4.27-2-686  vmlinuz-2.4.27-3-686
```

initrd.img-・・・というファイルがInitrdです。ファイル形式をみてみます。ファイル形式をみるとときには、fileコマンドが便利です。

```
# file initrd.img-2.4.27-3-686
initrd.img-2.4.27-3-686: Linux Compressed ROM File System
data, little endian size 4243456
version #2 sorted_dirs CRC 0x8090125d, edition 0, 2246 blocks,
300 files
```

ふむ。Compressed ROM File System dataって書いてありますがcramfsのことです。私のPCはDebianでしたのでcramfsでしたが、他のディストリビューションの場合は、gzip+ext2の場合があるかもしれませんので、ご注意ください。

マウントしてみましょう。ループバックマウントなので「-o loop」を忘れずに。

```
# mount -t cramfs -o loop initrd.img-2.4.27-3-686 /mnt
```

/mnt にマウントしましたので、Initrdの中を探検してみましょう。

```
# cd /mnt
# ls
bin  dev2  keyscripts  linuxrc.conf  proc  scripts  usr
bin2 devfs  lib         loadmodules   sbin  sys      var
dev  etc   linuxrc    mnt           script tmp
```

普通のファイルシステムですね!

では、デバイスドライバでもみてみますか～。

```
# cd lib/modules/2.4.27-3-686/
# ls
initrd          modules.generic_string  modules.parpportmap  modules.usbmap
kernel          modules.ieee1394map     modules.pcimap
modules.dep     modules.isapnpmap      modules.pnpbiosmap
# cd kernel/drivers/
# ls
block cdrom ide md message parport pcmcia pnp scsi video
debian:/mnt/lib/modules/2.4.27-3-686/kernel/drivers# ls -R
.:
```

```
block cdrom ide md message parport pcmcia pnp scsi video

./block:
DAC960.o cciss.o cpqarray.o floppy.o ps2esdi.o sx8.o umem.o xd.o

./cdrom:
cdrom.o

./ide:
ide-cd.o ide-detect.o ide-floppy.o legacy raid
ide-core.o ide-disk.o ide-pnp.o pci

./ide/legacy:
ide-cs.o

./ide/pci:
adma100.o cmd640.o hpt34x.o pdc202xx_old.o siimage.o via82cxxx.o
aec62xx.o cmd64x.o hpt366.o piix.o sis5513.o
alim15x3.o cs5530.o ns87415.o rz1000.o slc90e66.o
amd74xx.o cy82c693.o opti621.o sc1200.o triflex.o
. . . . .
. . . . .
(以下、長いので略)
```

へえ、こんなに沢山あるんだ・・。知らなかった。