华清远见 FARSIGHT 嵌入式培训专家

嵌入式学院 华清远见旗下品牌

高效学习Linux内核

任桥伟

- 把内核当朋友
- 先会使用它
- 依照4个层次进行内核学习
- 走出心理误区
- 使用vim+cscope+ctags浏览内核源码
- 使用Kernel地图定位目标代码
- 分析内核源码：态度决定一切
- 以内核源码为中心，坚持学习资源建设

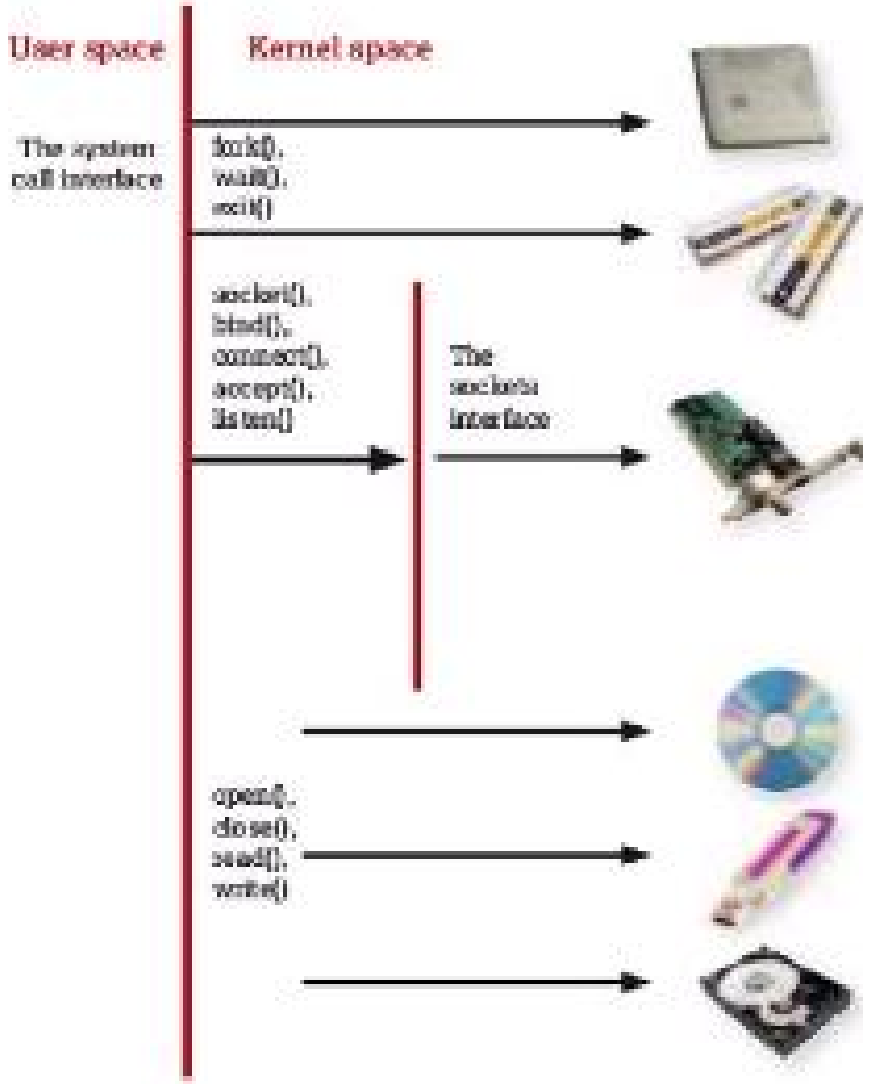
Kernel基本信息

- 某个东西的核心部分
- Linus等写的那点儿代码
- 那点儿代码是复杂得
- Kernel的成人礼
- 多变的版本号
  - X.Y.Z
- Kernel与那些发行版

内核的体系结构

Kernel是如何工作的

- “Linux”虚拟机——系统调用
- 台前——文件系统
- 幕后——进程调度
- 幕后——内存管理
- 模块化的Kernel



如何选择发行版

- 发行版的风云排行榜
- 发行版PK服务
- 一些问卷
- 别人的意见
- 重要的是开始去使用

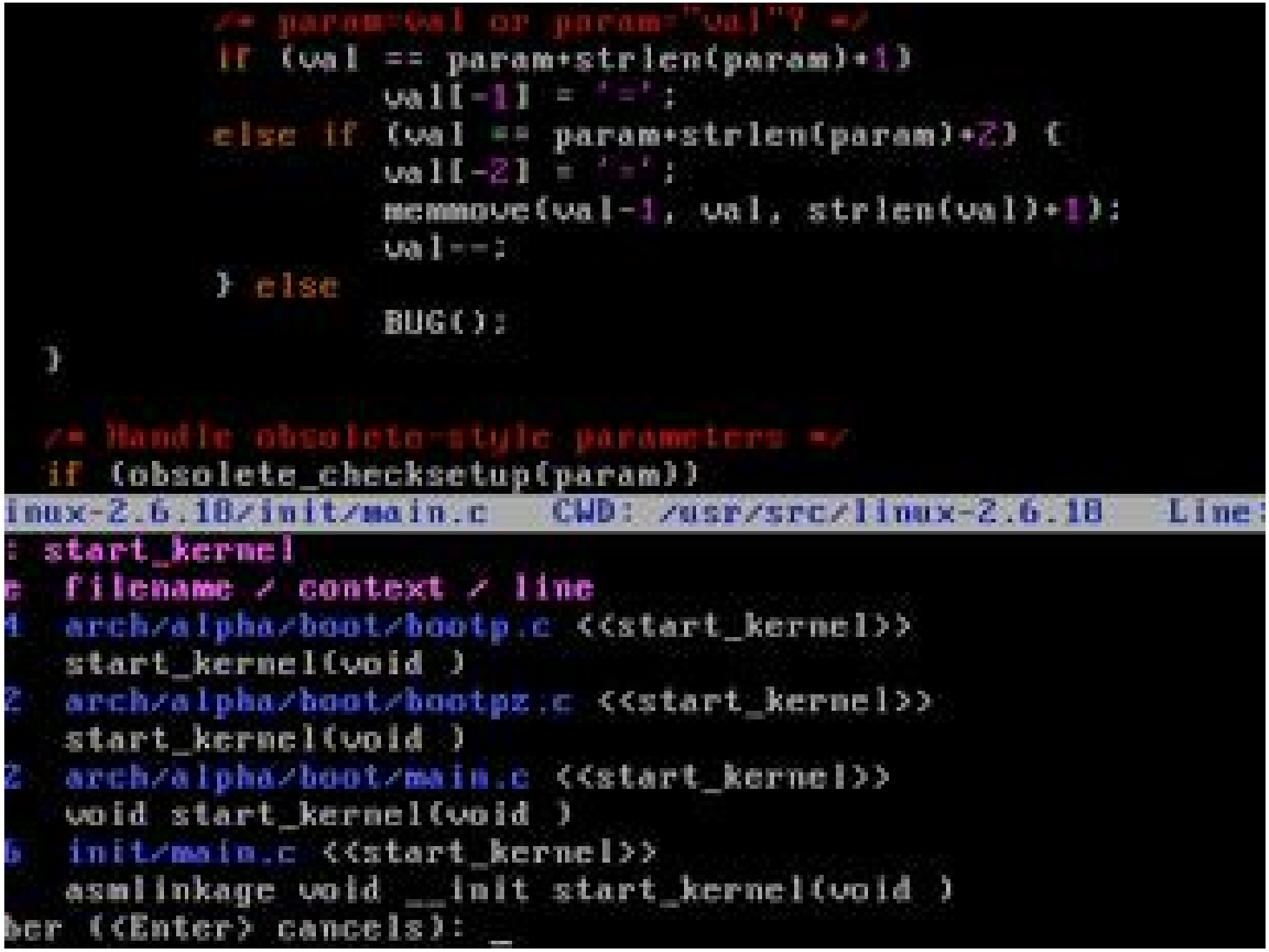
Kernel的修炼之道

- 炼气期——全面了解抓基本
- 筑基期——兴趣导向深钻研
- 结丹期——融入社区做贡献
- 元婴期——坚持坚持再坚持

内核学习的心理误区

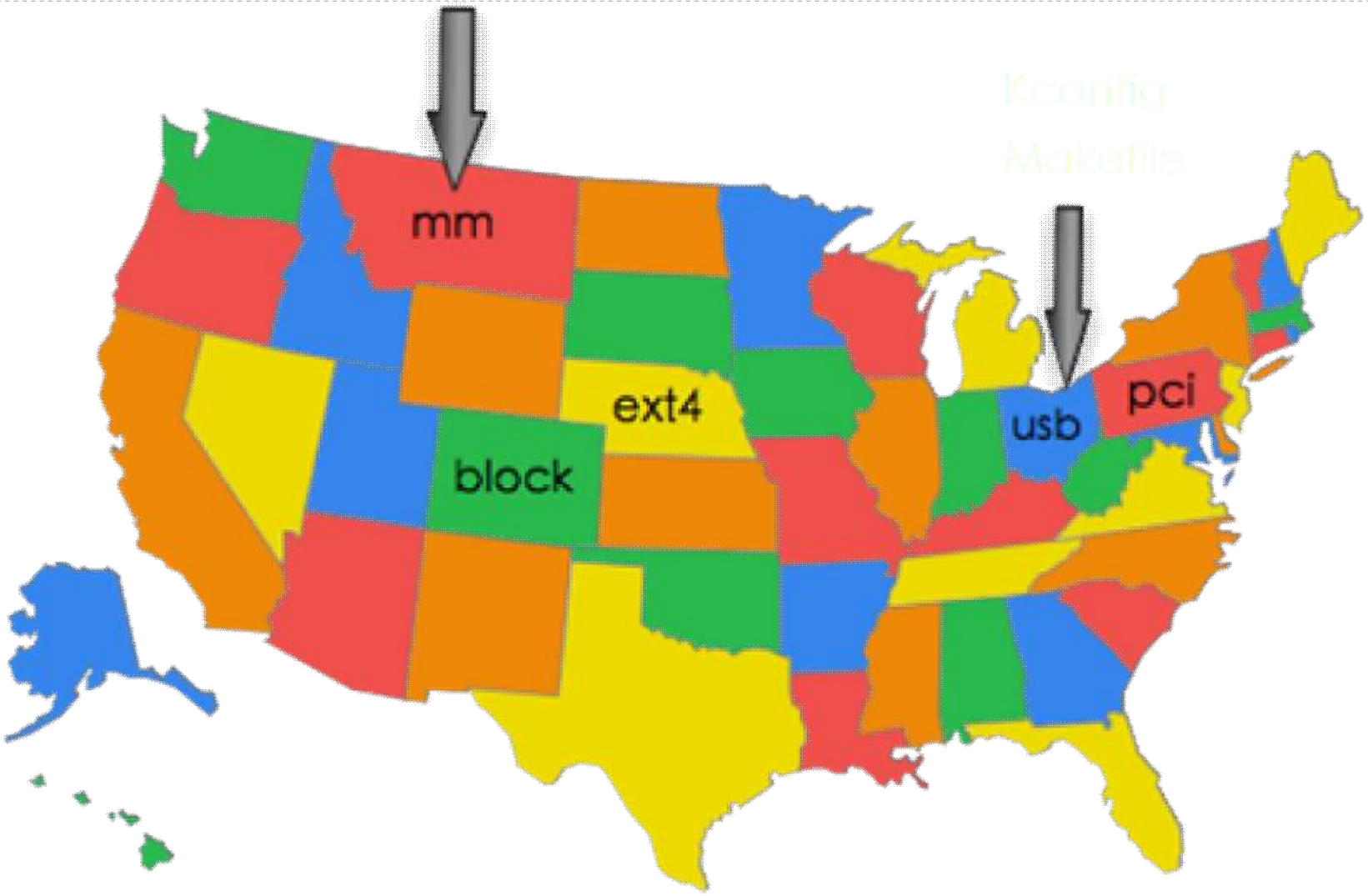
- 内核学习的效果取决于两个方面
  - 心理
  - 方法论
- 两个最主要的心理问题
  - 盲从
  - 恐惧

使用vim+cscope+ctags



Kernel地图

- 学习内核就是学习内核源代码
- **Kernel地图很好很强大**
  - **Kconfig**
  - **Makefile**
- 能够利用**Kernel**地图去定位目标代码
  - 一个**U**盘驱动的例子


mm
ext4
block
usb
pci

drivers/usb/storage/Kconfig

```
101 config USB_STORAGE_SDDR55
102     bool "SanDisk SDDR-55 SmartMedia support (EXPERIMENTAL)"
103     depends on USB_STORAGE && EXPERIMENTAL
104     help
105       Say Y here to include additional code to support the Sandisk SDDR-55
106       SmartMedia reader in the USB Mass Storage driver.
```

```
34 config USB_STORAGE_DATAFAB
35     bool "Datafab Compact Flash Reader support (EXPERIMENTAL)"
36     depends on USB_STORAGE && EXPERIMENTAL
37     help
38       Support for certain Datafab CompactFlash readers.
39       Datafab [illegible]
```

```
 9 config USB_STORAGE
10  tristate "USB Mass Storage support"
11      depends on USB && SCSI
12  ---help---
13    Say Y here if you want to connect USB mass storage devices to your
14    computer's USB port. This is the driver you need for USB
15    floppy drives, USB hard disks, USB tape drives, USB CD-ROMs,
16    USB flash devices, and memory sticks, along with
17    similar devices. This driver may also be used for some cameras
18    and card readers.
19
20    This option depends on 'SCSI' support being enabled, but you
21       probably also need 'SCSI device support: SCSI disk support'
22    (BLK_DEV_SD) for most USB storage devices.
23
24    To compile this driver as a module, choose M here: the
25    module will be called usb-storage.
```

drivers/usb/storage/Makefile

```
 7 EXTRA_CFLAGS := -Idrivers/scsi
 8
 9 obj-$(CONFIG_USB_STORAGE)   += usb-storage.o
10
11 usb-storage-obj-$(CONFIG_USB_STORAGE_DEBUG) += debug.o
12 usb-storage-obj-$(CONFIG_USB_STORAGE_USBAT) += shuttle_usbat.o
13 usb-storage-obj-$(CONFIG_USB_STORAGE_SDDR09)    += sddr09.o
14 usb-storage-obj-$(CONFIG_USB_STORAGE_SDDR55)    += sddr55.o
15 usb-storage-obj-$(CONFIG_USB_STORAGE_FREECOM)   += freecom.o
16 usb-storage-obj-$(CONFIG_USB_STORAGE_DPCM)  += dpcm.o
17 usb-storage-obj-$(CONFIG_USB_STORAGE_ISD200)    += isd200.o
18 usb-storage-obj-$(CONFIG_USB_STORAGE_DATAFAB)   += datafab.o
19 usb-storage-obj-$(CONFIG_USB_STORAGE_JUMPSHOT)  += jumpshot.o
20 usb-storage-obj-$(CONFIG_USB_STORAGE_ALAUDA)    += alauda.o
21 usb-storage-obj-$(CONFIG_USB_STORAGE_ONETOUCH)  += onetouch.o
22 usb-storage-obj-$(CONFIG_USB_STORAGE_KARMA) += karma.o
23
24 usb-storage-objs := scsiglue.o protocol.o transport.o usb.o \
25         initializers.o $(usb-storage-obj-y)
26
27 ifneq ($(CONFIG_USB_LIBUSUAL),)
28  obj-$(CONFIG_USB)   += libusual.o
29 endif
```

分析内核源码如何入手

- 阅读README
- 利用Kernel地图
  - 分析Kconfig
  - 分析Makefile
- 从初始化函数开始分析
  - 态度决定一切

drivers/usb/README


```
atm  class  co
Makefile  READ
```

```
 23 Here is a list of what each subdirectory here is, and what is contained
in
 24 them.
 25
 26 core/        - This is for the core USB host code, including the
 27             usbfs files and the hub class driver ("khubd").
 28
 40 image/       - This is for still image drivers, like scanners or
 41             digital cameras.
 42 input/       - This is for any driver that uses the input subsystem,
 43             like keyboard, mice, touchscreens, tablets, etc.
 44 media/       - This is for multimedia drivers, like video cameras,    and
 45             radios, and any other drivers that talk to the v4l
 46             subsystem.
 47 net/     - This is for network drivers.
 48 serial/      - This is for USB to serial drivers.
 49 storage/ - This is for USB mass-storage drivers.
 50 class/       - This is for all USB device drivers that do not fit
 51             into any of the above categories, and work for a range
 52             of USB Class specified devices.
 53 misc/        - This is for all USB device drivers that do not fit
 54             into any of the above categories.
```

drivers/usb/core/Kconfig

```
15 config USB_DEVICEFS
16         bool "USB device filesystem"
17         depends on USB
18         ---help---
19           If you say Y here (and to "/proc file system support" in the "File
20                     systems"  section,  above),  you  will  get  a  file
/proc/bus/usb/devices
21           74 config USB_SUSPEND
22           75       bool "USB selective suspend/resume and wakeup (EXPERIMENTAL)"
23           76       depends on USB && PM && EXPERIMENTAL
24           77       help
25           78         If you say Y here, you can use driver calls or the sysfs
26           79         "power/state" file to suspend or resume individual USB
27           80         peripherals.
28           81
29           82         Also, USB "remote wakeup" signaling is supported, whereby some
             83         USB devices (like keyboards and network adapters) can wake up
             84         their parent hub.  That wakeup cascades up the USB tree, and
             85         could wake the system from states like suspend-to-RAM.
             86
             87         If you are unsure about this, say N here.
```

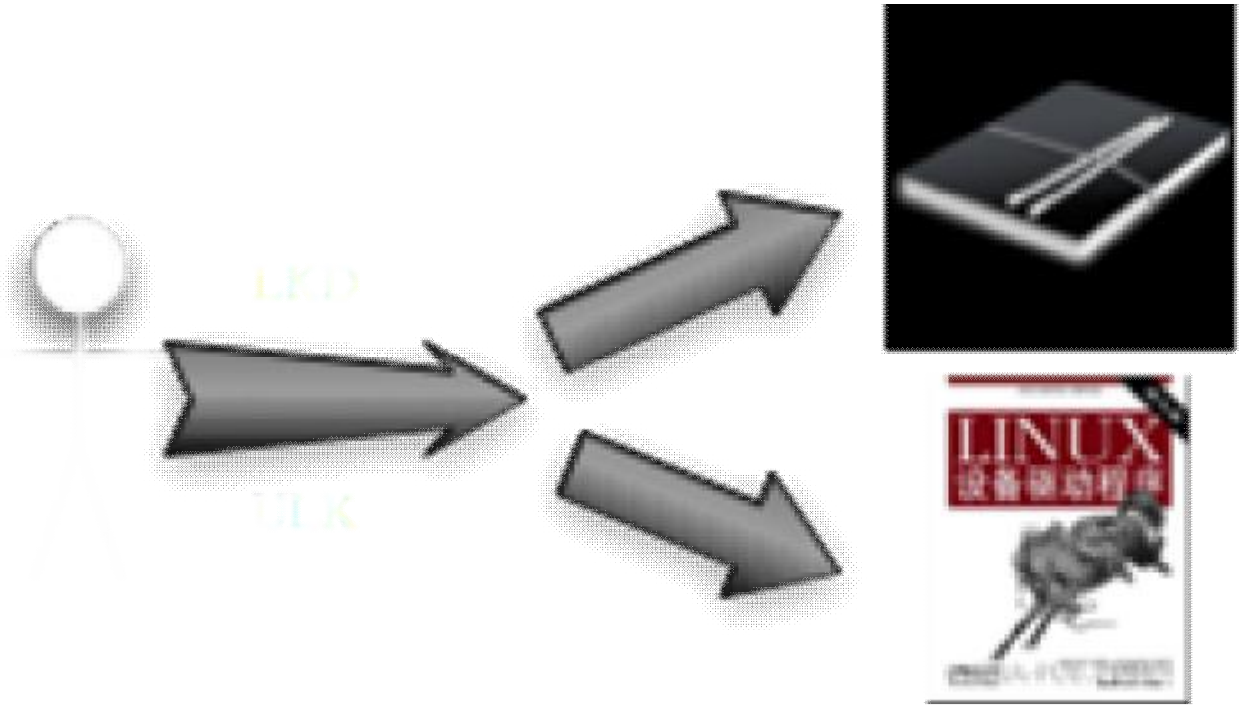
drivers/usb/core/Makefile

```
 5 usbcore-objs  := usb.o hub.o hcd.o urb.o message.o driver.o \
 6                  config.o file.o buffer.o sysfs.o endpoint.o \
 7                  devio.o notify.o generic.o quirks.o
 8
 9 ifeq ($(CONFIG_PCI),y)
10        usbcore-objs    += hcd-pci.o
11 endif
12
13 ifeq ($(CONFIG_USB_DEVICEFS),y)
14        usbcore-objs    += inode.o devices.o
15 endif
16
17 obj-$(CONFIG_USB)       += usbcore.o
18
19 ifeq ($(CONFIG_USB_DEBUG),y)
20 EXTRA_CFLAGS += -DDEBUG
21 endif
```

初始化函数usb_init()

```
865 static int __init usb_init(void)
866 {
867     int retval;
868     if (nousb) {
869         pr_info("%s: USB support disabled\n", usbcore_name);
870         return 0;
871     }
872
873     retval = ksuspend_usb_init();
874     if (retval)
875         goto out;
876     retval = bus_register(&usb_bus_type);
877     if (retval)
878         goto bus_register_failed;
879     retval = usb_host_init();
880     if (retval)
881         goto host_init_failed;
```

如何利用经典书籍



其他相关资源

- 内核文档
- 社区
  - **lkml**
  - 中文邮件列表
- 网站或论坛

Q&A

我的联系方式：

Email： rqw@farsight.com.cn

Blog：blog.csdn.net/fudan_abc

谢谢！