Linux进程、线程和调度(4)

讲解时间：5月22-25日晚9点

宋宝华

麦当劳喜欢您来，喜欢您再来

扫描关注

Linuxer

第四次课大纲

1. 多核下负载均衡
2. 中断负载均衡、RPS软中断负载均衡
3. cgroups和CPU资源分群分配
4. Android和Docker对cgroup的采用
5. Linux为什么不是硬实时的
6. preempt-rt对Linux实时性的改造

练习题

1. 用time命令跑1个含有2个死循环线程的进程
2. 用taskset调整多线程依附的CPU
3. 创建和分群CPU的cgroup，调整权重和quota
4. cyclictest

负载均衡

- RT 进程：N个优先级最高的RT分布到N个核
  - pull_rt_task()
  - push_rt_task()

- 普通进程
  - 周期性负载均衡
  - IDLE时负载均衡
  - fork和exec时负载均衡

CPU task affinity

- 设置affinity

```
int pthread_attr_setaffinity_np(pthread_attr_t *, size_t, const cpu_set_t *);
int pthread_attr_getaffinity_np(pthread_attr_t *, size_t, cpu_set_t *);
int sched_setaffinity(pid_t pid, unsigned int cpusetsize, cpu_set_t *mask);
int sched_getaffinity(pid_t pid, unsigned int cpusetsize, cpu_set_t *mask);
```

taskset

- taskset -a -p 01 19999
- taskset -a -p 02 19999
- taskset -a -p 03 19999

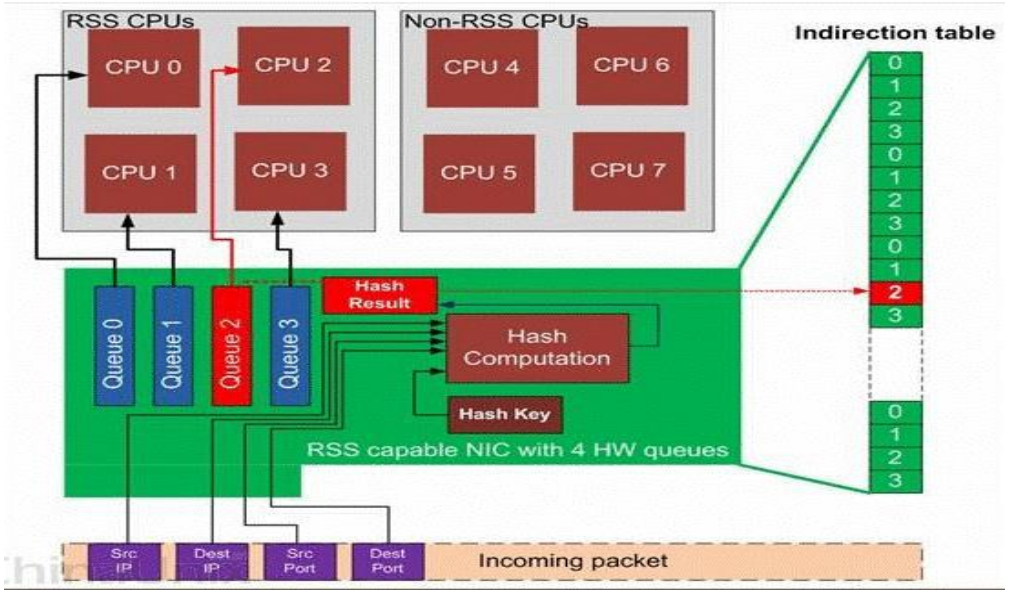
IRQ affinity

- 分配IRQ到某个CPU

[root@boss ~]# echo 01 > /proc/irq/145/smp_affinity
[root@boss ~]# cat /proc/irq/145/smp_affinity
00000001

- mq ethernet

/proc/irq/74/smp_affinity 000001
/proc/irq/75/smp_affinity 000002
/proc/irq/76/smp_affinity 000004
/proc/irq/77/smp_affinity 000008
…



多核间的softIRQ scaling

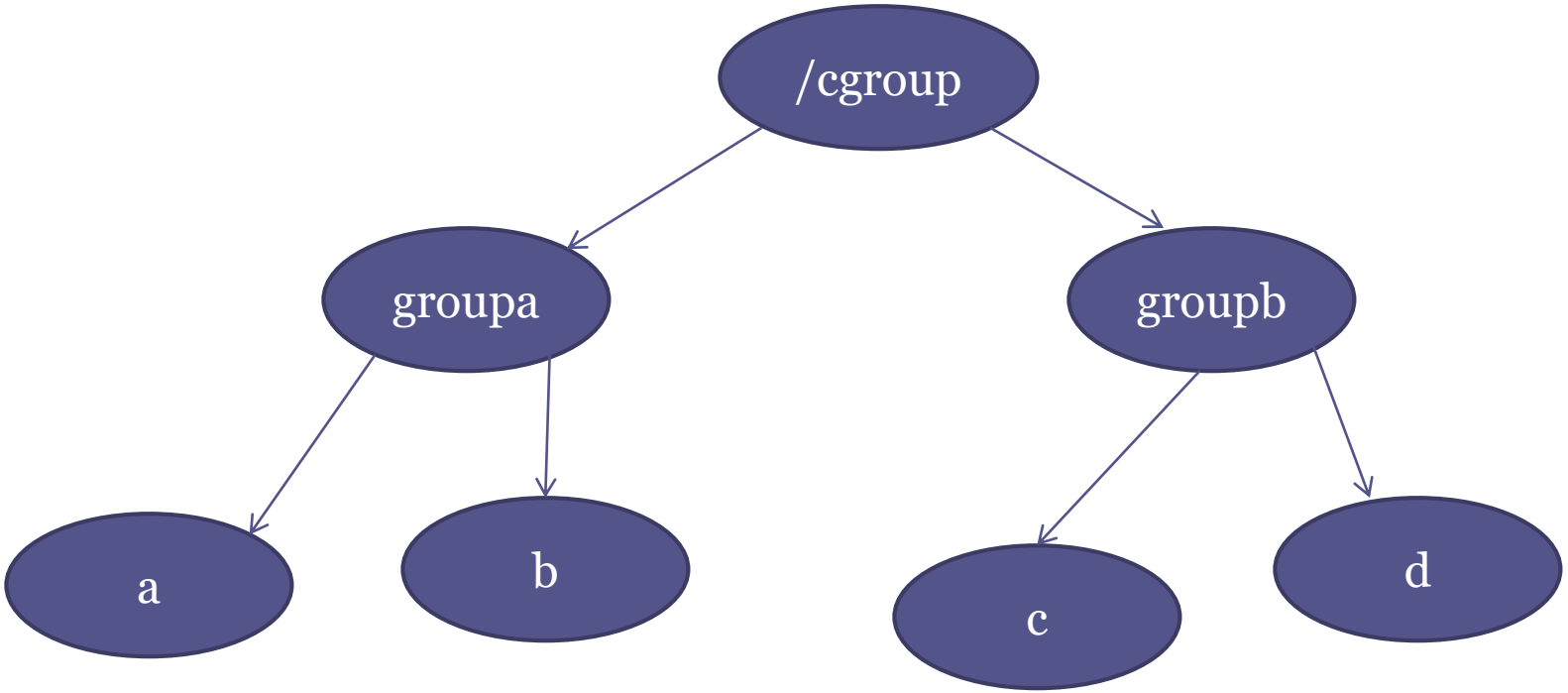
- RPS 将包处理负载均衡到多个CPU

#例如
[root@machine1 ~]# echo fffe > /sys/class/net/eth1/queues/rx-0/rps_cpus
fffe

#观察
[root@machine1 ~]# watch -d "cat /proc/softirqs | grep NET_RX"

cgroup

- 定义不同cgroup CPU分享的share
- 定义某个cgroup在某个周期里面最多跑多久



Android和cgroup

- apps, bg_non_interactive

Shares:

apps: cpu.shares = 1024

bg_non_interactive: cpu.shares = 52

Quota:

apps:

cpu.rt_period_us: 1000000 cpu.rt_runtime_us: 800000

bg_non_interactive:

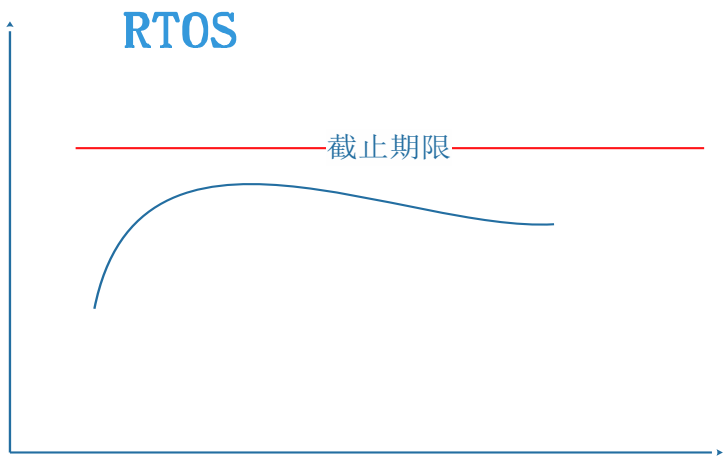
cpu.rt_period_us: 1000000 cpu.rt_runtime_us: 700000

Docker和cgroup

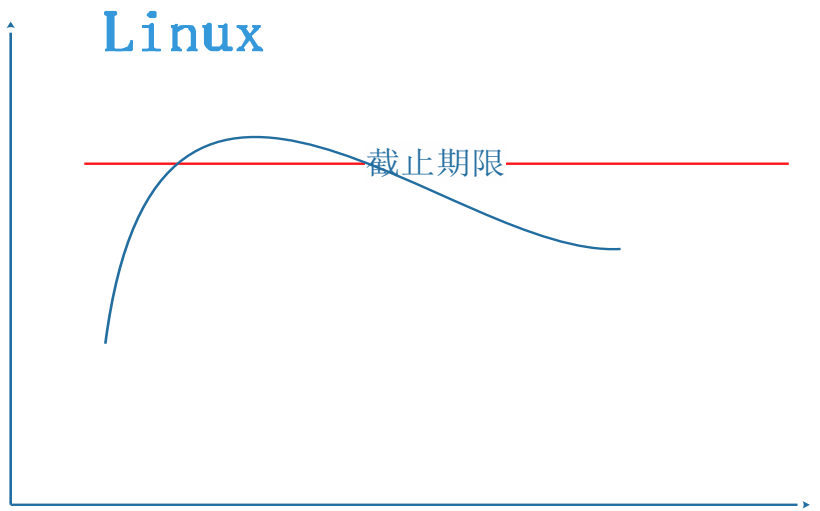
- Docker使用cgroup调配容器的CPU资源

```
$ docker run --cpu-quota 25000 --cpu-period 10000 --cpu-shares 30
linuxep/lepv0.1

baohua@ubuntu:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS              PORTS               NAMES
3f39ca25d14d

baohua@ubuntu:/sys/fs/cgroup/cpu/docker$ cd 3f39c...
baohua@ubuntu:/sys/fs/cgroup/cpu/docker/3f39c...$ ls
cgroup.clone_children  cgroup.procs  cpuacct.stat  cpuacct.usage
cpuacct.usage_percpu  cpu.cfs_period_us  cpu.cfs_quota_us  cpu.shares  cpu.stat
notify_on_release  tasks
baohua@ubuntu:/sys/fs/cgroup/cpu/docker/3f39c...$ cat cpu.cfs_quota_us
25000
baohua@ubuntu:/sys/fs/cgroup/cpu/docker/3f39c...$ cat cpu.cfs_period_us
10000
baohua@ubuntu:/sys/fs/cgroup/cpu/docker/3f39c...$ cat cpu.shares
30
```

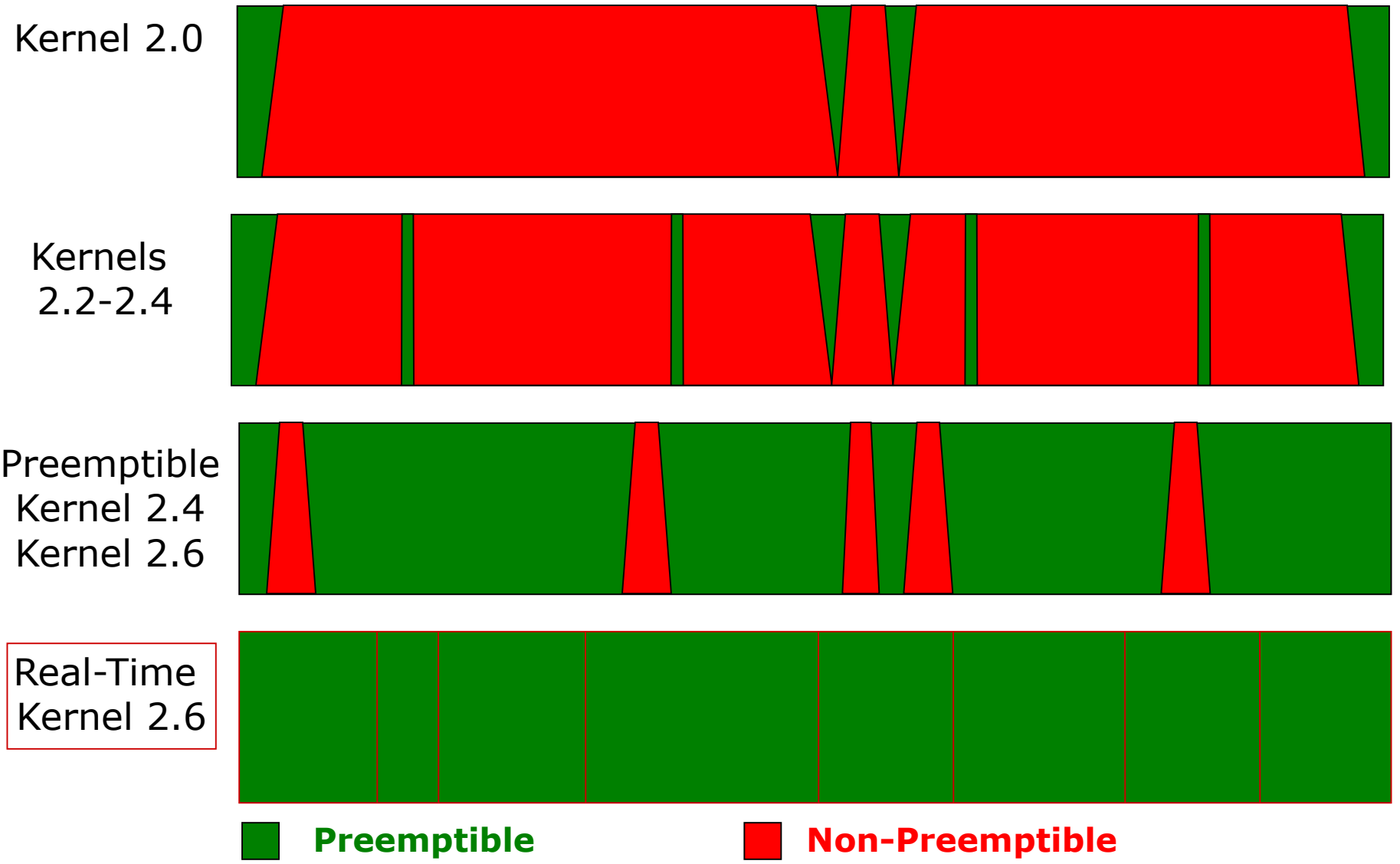
Hard realtime - 可预期性



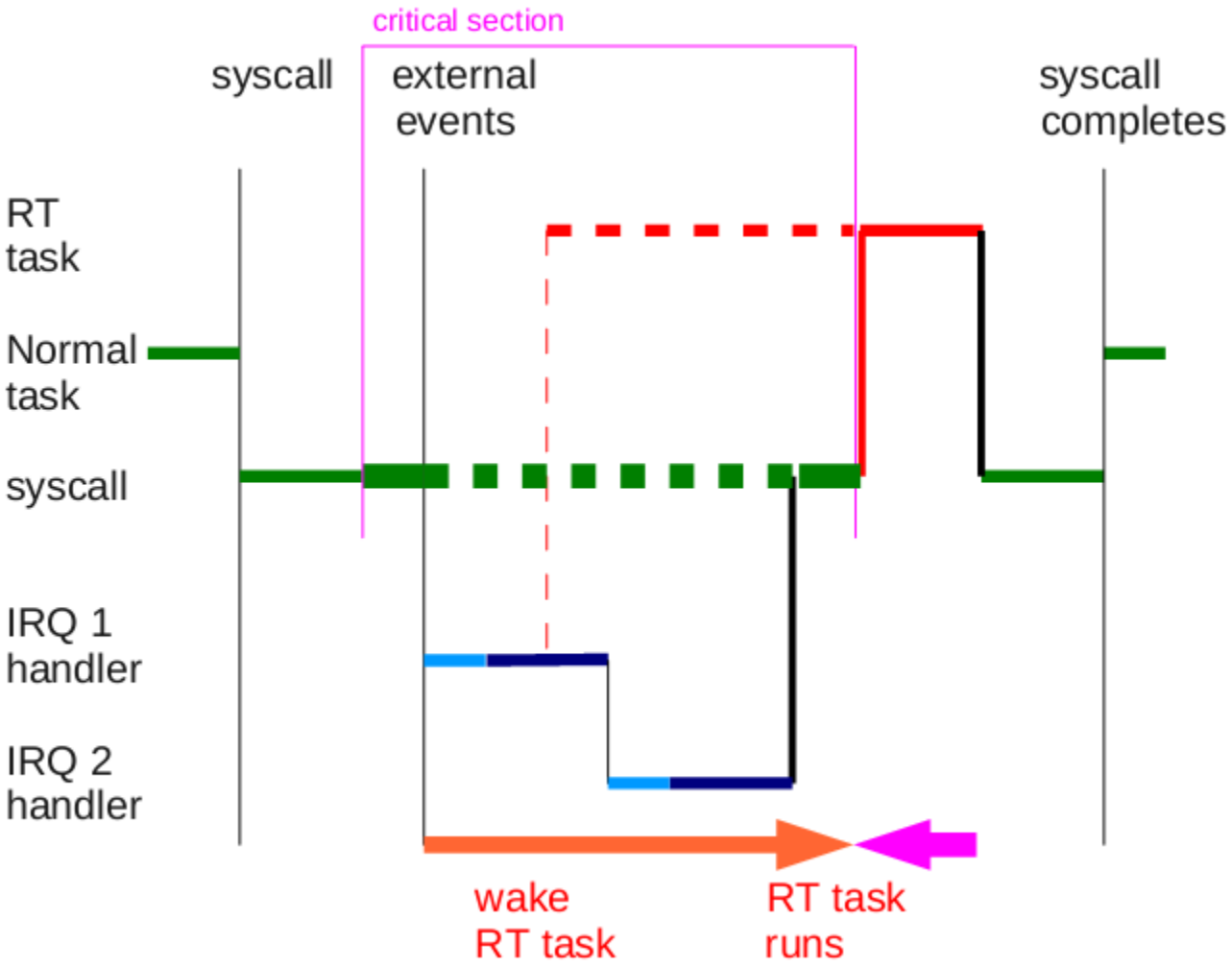
VS.



Kernel越发支持抢占

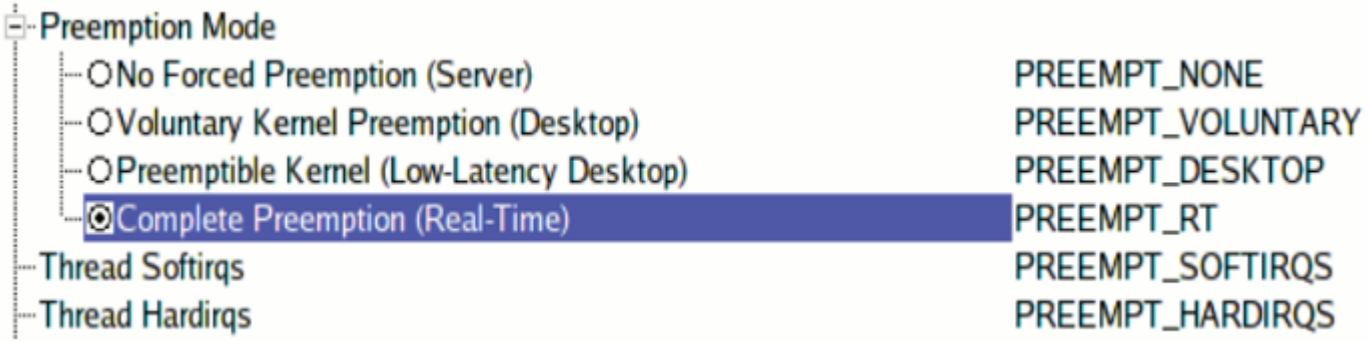


Linux为什么不硬实时



PREEMPT_RT补丁

- spinlock迁移为可调度的mutex,同时报了raw_spinlock_t
- 实现优先级继承协议
- 中断线程化
- 软中断线程化



课程练习源码

https://github.com/21cnbao/process-courses

谢 谢！