

# 迅为电子Linux驱动教程 ——内核开发基础

北京迅为电子有限公司



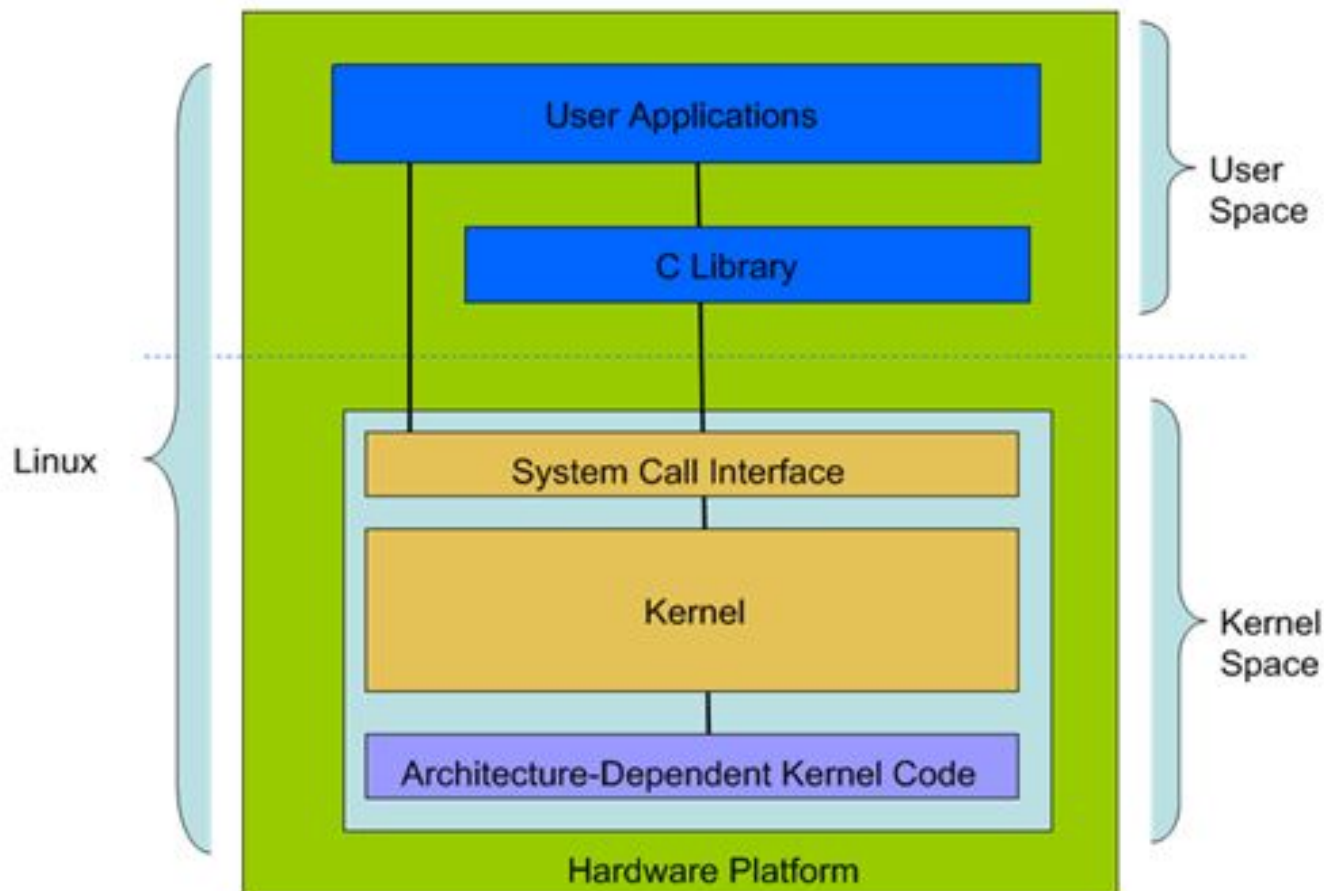


# 主要内容

- Linux体系结构
- Linux内核结构
- Linux内核源码目录结构



# Linux体系结构





# Linux体系结构

- 从上图可知，Linux体系结构由用户空间和内核空间构成
- 为什么Linux体系要分为用户空间和内核空间？
- 从程序员的角度分析
  - 将linux底层和应用分开，做应用的做应用，做底层的做底层，各干各的。  
经济学的原理是，分工产生效率。

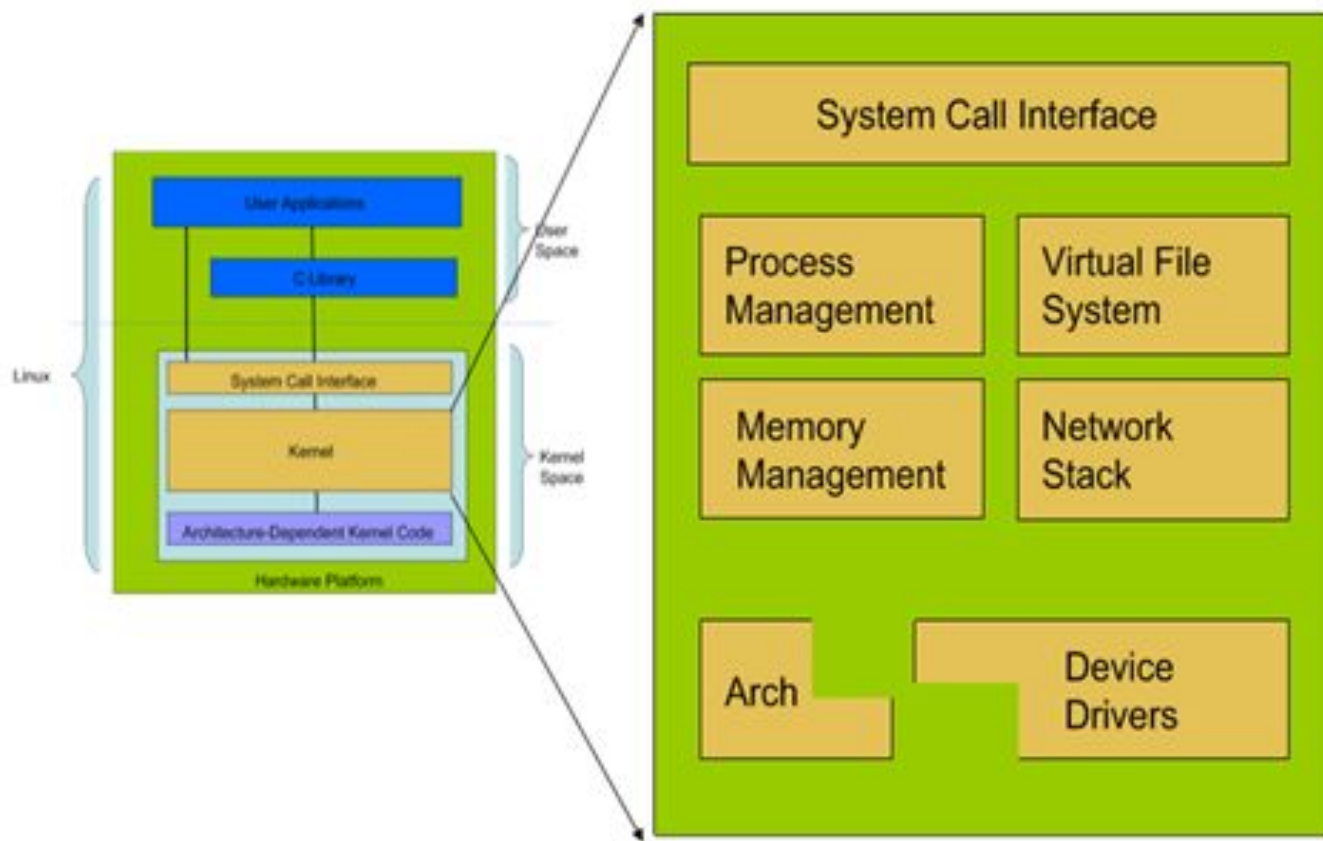


# Linux体系结构

- 从安全性的角度分析，为了保护内核。现代CPU通常都实现了不同的工作模式。以ARM为例：ARM实现了7种工作模式，不同模式下CPU可以执行的指令或者访问的寄存器不同：
  - (1)用户模式 usr (2)系统模式 sys(3)管理模式 svc(4)快速中断 fiq(5)外部中断 irq(6)数据访问终止 abt(7)未定义指令异常
  - 以(2)X86为例：X86实现了4个不同级别的权限，Ring0—Ring3 ;Ring0下可以执行特权指令，可以访问IO设备；Ring3则有很多的限制
  - 以Android为例：将应用放到Java虚拟机上面运行，应用更加远离底层
- 用户空间和内核空间是程序执行的两种不同状态，我们可以通过“系统调用”和“硬件中断”来完成用户空间到内核空间的转移



# Linux内核结构





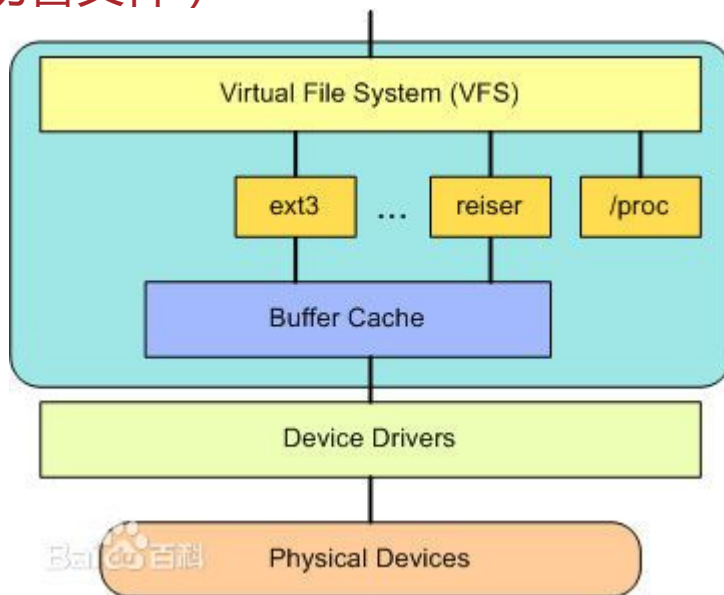
# Linux内核结构

- System Call Interface ( SCI层 )
  - 为用户空间提供了一套标准的系统调用函数来访问Linux内核。
- Procees Management ( PM )
  - 进程管理是创建进程 ( fork、exec ) ,停止进程 ( kill、exit ) ,并控制他们之间的通信 ( signal等 ) 。还包括进程调度 , 控制活动进程如何共享CPU
- Memory Management ( MM )
  - 内存管理的主要作用是控制多个进程安全的共享内存区域



# Linux内核结构

- Virtual File Systems ( VFS )
  - 虚拟文件系统，隐藏各种文件系统的具体细节，为文件操作提供统一的接口
  - Linux提供了一个大的通用模型，使这个模型包含了所有文件系统功能的集合（一切皆文件）







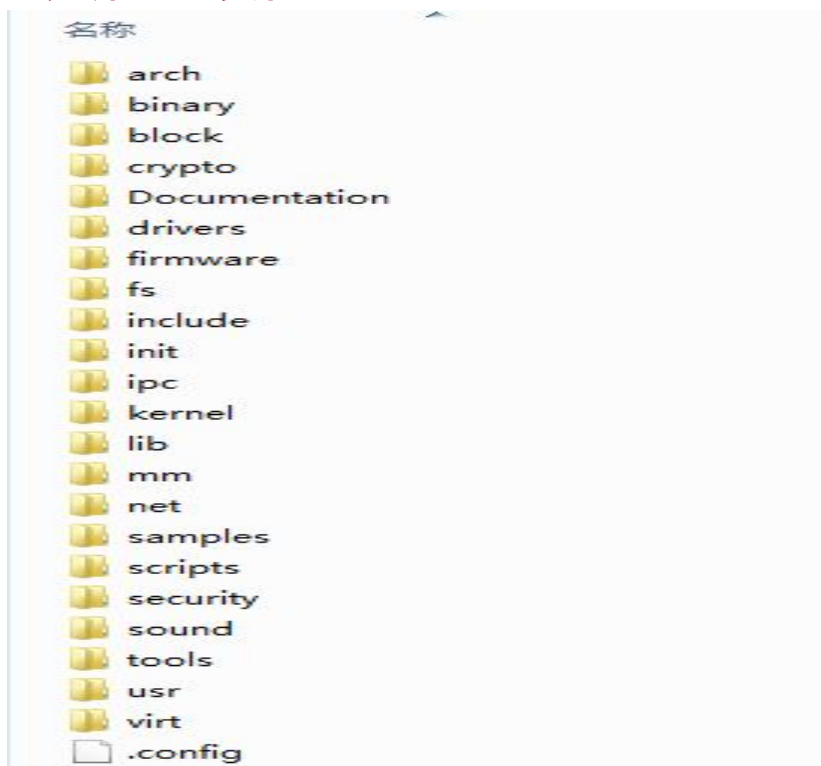
# linux内核结构

- Device Drivers设备驱动
  - Linux内核中有大量的代码在设备驱动程序部分，用于控制特定的硬件设备。
  - Linux驱动一般分为网络设备、块设备、字符设备、杂项设备
- 网络协议栈
  - 内核网络协议栈为Linux提供了丰富的网络协议实现。



# 目录结构

- Linux内核源码采用树形结构。功能相关的文件放到不同的子目录下面，使程序更具有可读行。



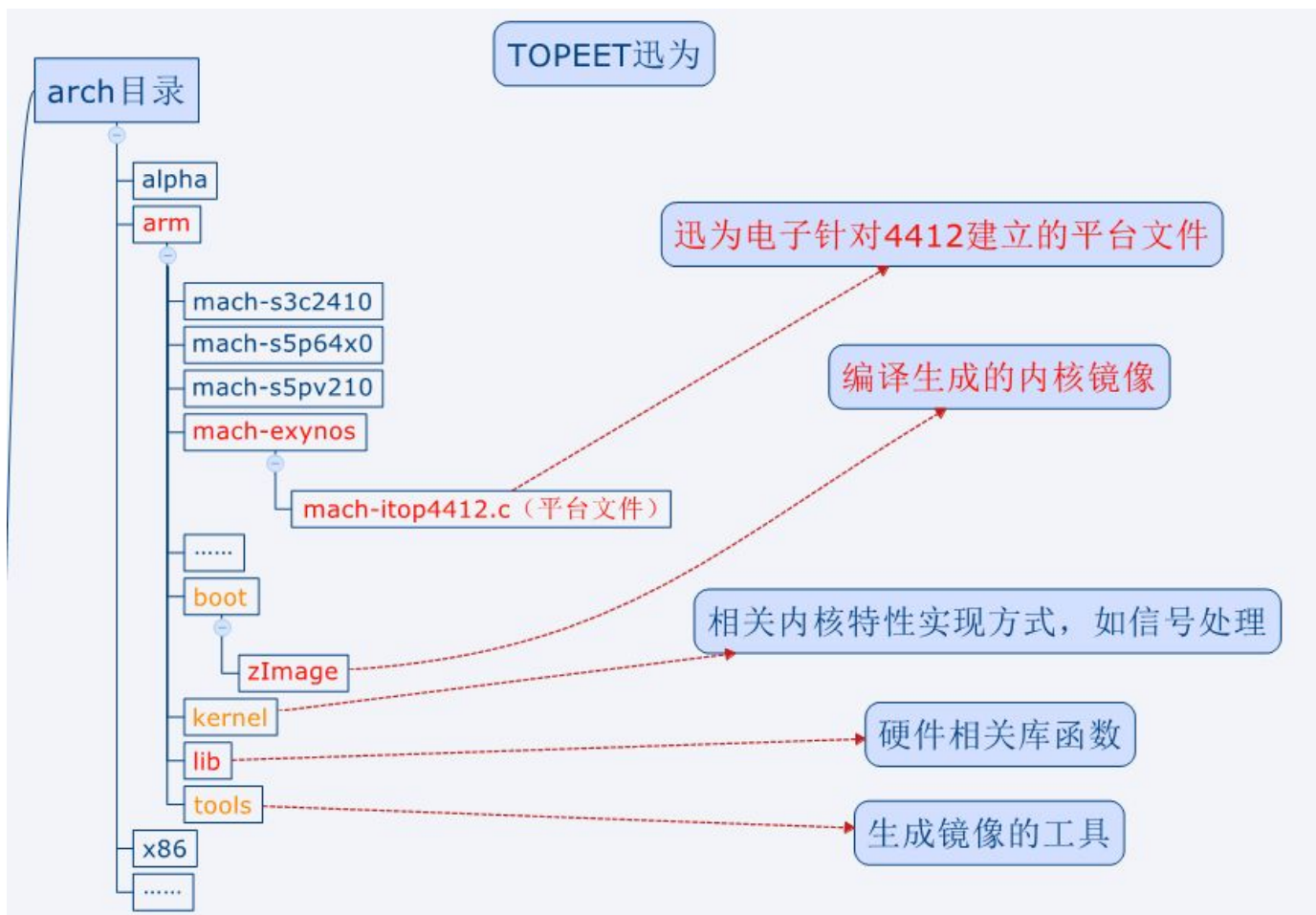


# 目录结构

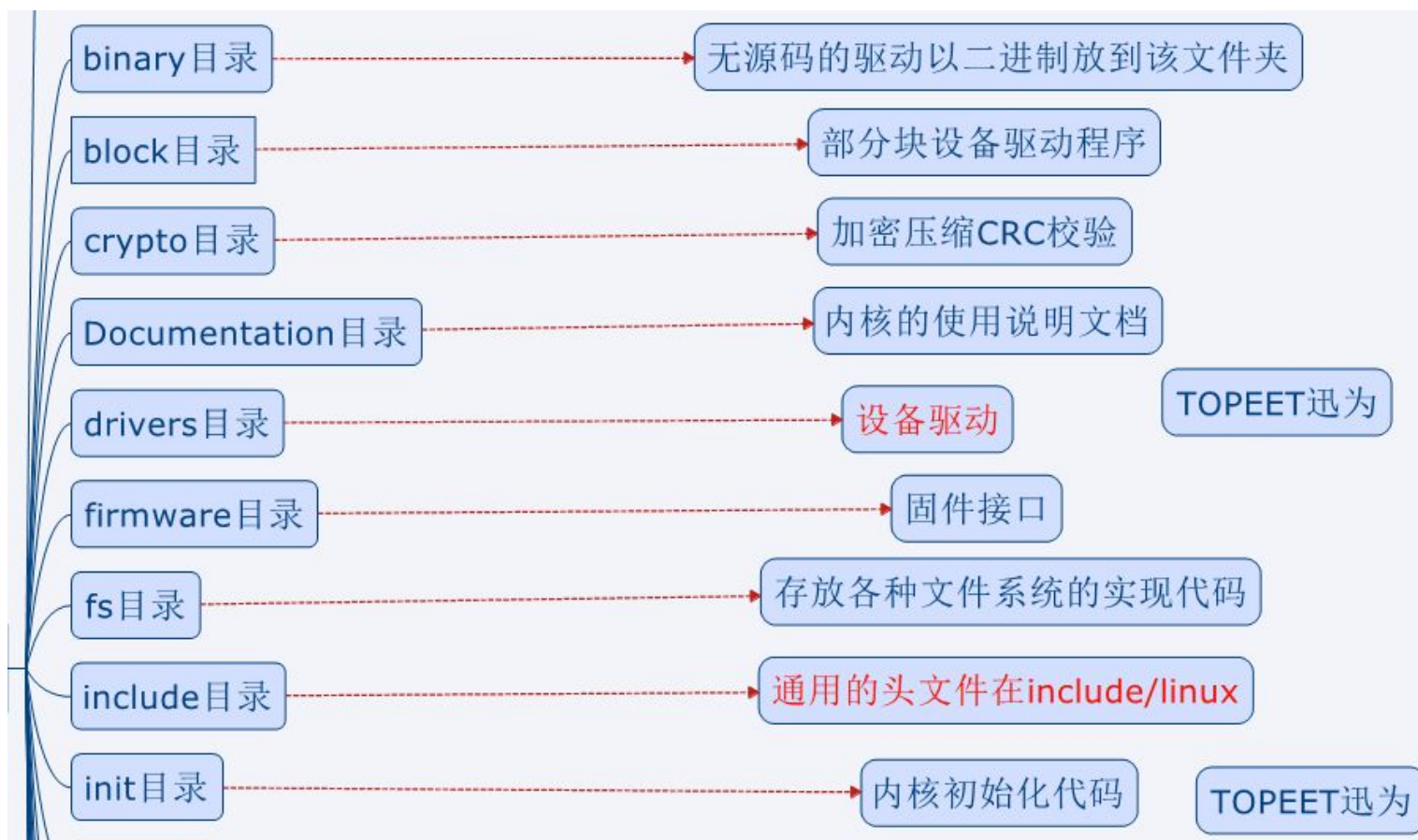
- arch目录
  - arch目录是平台目录。内核支持的所有CPU架构，在该目录下都有对应的子目录。每个CPU的子目录，又进一步分解为boot，mm，kernel等子目录，分别控制系统引导，内存管理，系统调用。还有动态调频，主频率设置部分等



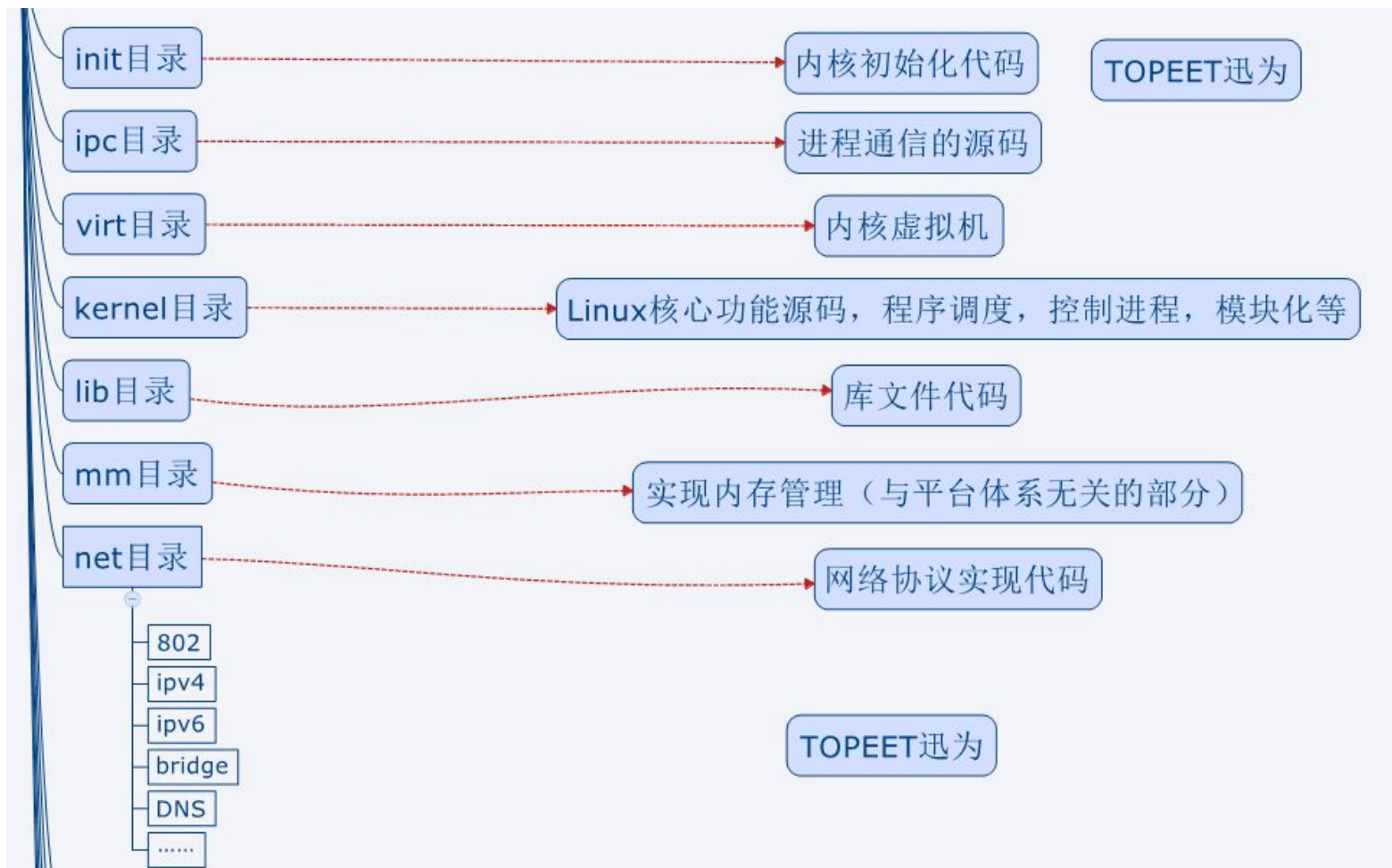
# 目录结构



# 目录结构



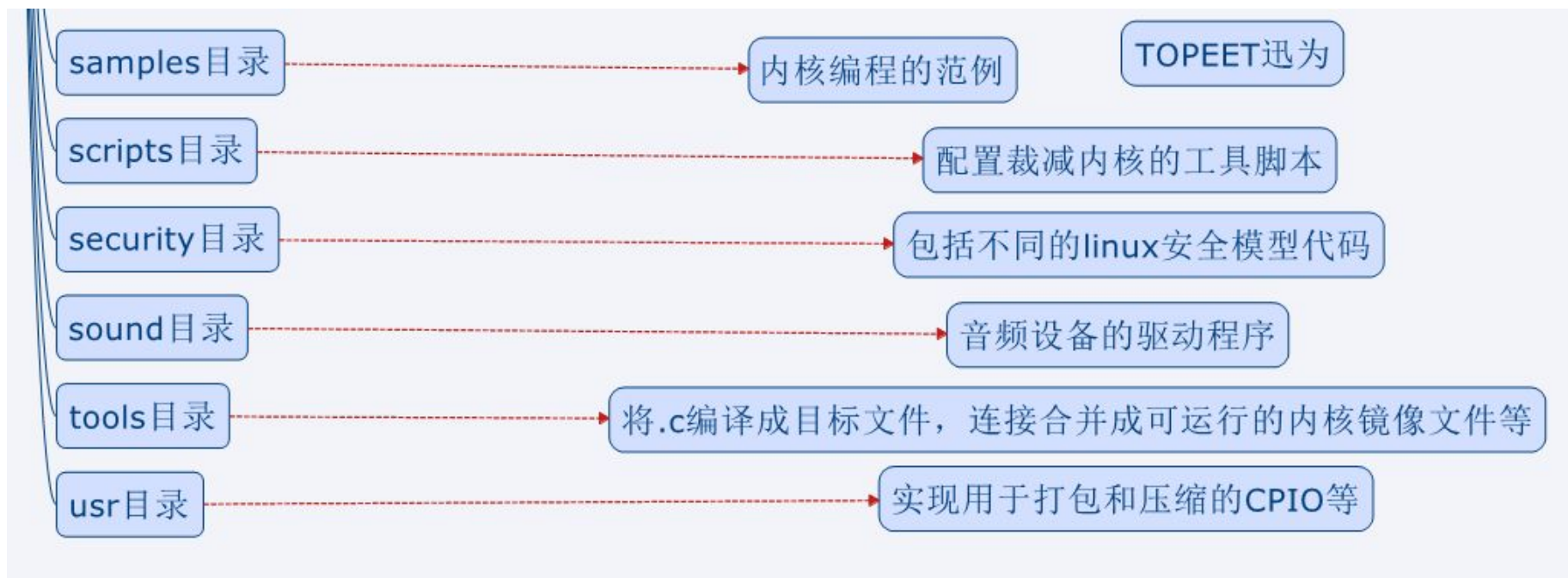
# 目录结构







# 目录结构





## 小结

- 2%的程序员写出80%的有效代码
  - 刚开始学习的时候最重要的就是看代码，但是学习Linux系统，不能一头扎进内核中无关的代码中，好几万个文件一辈子也看不完。
- Linux很复杂，很强大。
  - 内核源码解压编译后有1.3G，通过裁减编译出的linux镜像zImage不到4M
  - 4M源码镜像中，还有大部分不用我们关心
  - 我们只需要了解对写代码有帮助的部分，掌握我们需要掌握的代码
  - 前面Linux代码中只有红色标记的一部分是需要了解或者掌握的
- 作为开发者只需要学会使用它，我们要重点学习只是其中很小的一部分。
  - 对架构有一定的了解即可
  - **学习的重点是驱动开发相关的知识、函数的调用、驱动的协议、简单的硬件知识**





谢谢！