



# 实时Linux的昨天、今天、明天

2009-9-23



# 版权

---

- } 华清远见嵌入式培训中心版权所有；
- } 未经华清远见明确许可，不能为任何目的以任何形式复制或传播此文档的任何部分；
- } 本文档包含的信息如有更改，恕不另行通知；
- } 保留所有权利。

# 知识点



- } 实时概念
- } 昨天
- } 今天
- } 明天
- } 各种实时Linux简介



# 什么是实时系统

---

## } 实时

就是一个特定任务的执行时间必须是确定的，可预测的，并且在任何情况下都能保证任务的时限（最大执行时间限制）。实时分软实时和硬实时。

## } 软实时

就是对任务执行时限的要求不那么严苛，即使在一些情况下不能满足时限要求，也不会对系统本身产生致命影响，例如，媒体播放系统，手机

## } 硬实时

就是对任务的执行时限的要求非常严格，无论在什么情况下，任务的执行实现必须得到绝对保证，否则将产生灾难性后果，如汽车控制系统，飞行导航

# 关键技术指标

---

## } 中断延迟

中断延迟就是从一个外部事件发生到相应的中断处理函数的第一条指令开始执行所需要的时间。很多实时任务是靠中断驱动的，而且中断事件必须在限定的时限内处理，否则将产生灾难性后果，因此中断延迟对于实时系统来说，是一个非常重要的指标。

## } 抢占延迟

有时也称调度延迟，抢占延迟就是从一个外部事件发生到相应的处理该事件的任务的第一条命令开始执行的时间。大多数实时系统都是处理一些周期性的或非周期性的重复事件，事件产生的频度就确定了任务的执行时限，因此每次事件发生时，相应的处理任务必须及时响应处理，否则将无法满足时限。抢占延迟就反映了系统的响应及时程度。

## 其他指标

---

- } 强负载下的稳定性
- } 开发工具
- } 可移植性
- } 可裁剪性
- } 支持的硬件平台
- } 实时应用程序
- } 实时UI

<http://www.embedu.org/Column/Column130.htm>

## 昨天

---

直接使用Linux标准内核，缺少优化或者自己随意裁剪，可移植性差，应用程序不丰富，不标准化

### } 内核不可抢占

在Linux 2.4和以前的版本，内核是不可抢占的，也就是说，如果当前任务运行在内核态，即使当前有更紧急的任务需要运行，当前任务也不能被抢占。因此那个紧急任务必须等到当前任务执行完内核态的操作返回用户态后或当前任务因需要等待某些条件满足而主动让出CPU才能被考虑执行，这很明显严重影响抢占延迟。

### } 中断关闭

Linux在一些同步操作中使用了中断关闭指令，中断关闭将增大中断延迟，降低系统的实时性。

## } 自旋锁 (spinlock)

自旋锁是在可抢占内核和SMP情况下对共享资源的一种同步机制，一般地一个任务对共享资源的访问是非常短暂的，如果两个任务竞争一个共享的资源时，没有得到资源的任务将自旋以等待另一个任务使用完该共享资源。这种锁机制是非常高效的，但是在保持自旋锁期间将失效抢占，这意味着抢占延迟将增加。

## } 中断总是最高优先级的

在Linux中，中断（包括软中断）是最高优先级的，不论在任何时刻，只要产生中断事件，内核将立即执行相应的中断处理函数以及软中断，等到所有挂起的中断和软中断处理完毕有才执行正常的任务。因此在标准的Linux系统上，实时任务根本不可能得到实时性保证。

# 昨天

---

## } 调度算法和调度点

在Linux 2.4和以前的版本，调度器的时间复杂度是 $O(n)$ 的，而且在SMP的情况下性能低，因为所有的CPU共享一个任务链表，任何时刻只能有一个调度器运行。因此，抢占延迟很大程度上来自于当前系统的任务数，具有非常大的不确定性和不可预测性。

## } **Linux 虚拟内存的可裁剪性，如uClinux**

# 今天

内核大幅优化，支持硬实时，出现为实时系统定制的图形界面及应用包

## } Linux支持内核可抢占模式

在Linux 2.6中，内核已经可以抢占，因而实时性得到了加强。但是内核中仍有大量的不可抢占区域，如由自旋锁（spinlock）保护的临界区，以及一些显式使用preempt\_disable失效抢占的临界区。

## } Linux增强中断下半部分tasklet, workqueue

增强对多处理器的支持，中断线程化，减少对实时任务的延迟

# 今天

---

## }Linux采用O(1)的调度算法

在2.6内核中引入的O(1)调度器很好地解决了这些问题。此外，即使内核是可抢占的，也不是在任何地方可以发生调度，例如在中断上下文，一个中断处理函数可能唤醒了某一高优先级进程，但是该进程并不能立即运行，因为在中断上下文不能发生调度，中断处理完了之后内核还要执行挂起的软中断，等它们处理完之后才有机会调度刚才唤醒的进程。在标准Linux内核中，调度点（有意安排的执行任务调度的点）并不多，对2.4和2.6内核测试的结果表明，缺乏调度点是影响Linux实时性的一个因素。

# 今天

---

- } WindRiver Linux
- } RT-Linux/RTAI
- } Montavista Linux
- } **Otopia(基于QT)**
- } **Maemo (基于GTK, Nokia table OS, N800)**

# 明天

---

内核优化几近完善，硬件加速逐渐普及，应用开发包越来越丰富，出现行业定制模板，标准接口，良好的开发环境

} 基于3D的UI: Clutter

} GLX -> XGL

} 标准接口

} 近Visual studio的友好开发环境

} Android

} Intel Moblin

# 明天

---

## Clutter

- } 快速创建生动多样的可移植的用户图形界面的工具  
基于OpenGL或者openGLES， 创建了一套控件及API
- } <http://www.clutter-project.org/>

# 明天

Movies

TV-Series

Music videos

Video clips



## American Beauty (1999)

1 | 4 ^ v

122min, Drama

Lester Burnham, a depressed suburban father in a mid-life crisis, decides to turn his hectic life around after developing an infatuation for his daughter's attractive friend.

# 明天



- } **Xgl** 是一个 **X server**，借由 **OpenGL** 技术充分发挥了新一代显卡优异的3D立体画面表现能力；利用了 **聚合视窗管理器** 的帮助（如 **Compiz** 或 **Beryl**）对所有使用X、OpenGL和 **XVideo** 所制作的应用和图形效果提供了 **硬件加速** 的支援
- } 能表现出包含调整视窗透明度、可旋转的3D桌面、果冻效果等等令人叹为观止的立体特效。
- } 这个计划最初由 **David Reveman** 发起，并在 **2006年1月2日** 的时候发布第一个版本

## 各种实时Linux简介

---

- } uCLinux
- } Arm-Linux
- } RT-Linux/RTAI
- } MontaVista Linux
- } WindRiver Linux
- } Nokia Table OS
- } Google Android
- } Intel Moblin

## $\mu$ Clinux

---

- }  $\mu$  Clinux <http://www.uclinux.org>
- }  $\mu$  Clinux的含义：  
**Micro-Control-Linux**: 专门针对微控制器领域而设计的嵌入式Linux系统
- }  $\mu$  Clinux同标准Linux的最大区别就在于内存管理
  - 1、标准Linux使用的虚拟存储器技术
  - 2、 $\mu$  Clinux针对NOMMU

## μ Clinux (2)

---

- } 减少内存分配开销，减少内存分配延迟
- } 应用程序直接面对物理内存，多进程支持麻烦
- } 外存交换空间无法使用，
- } 仍然采用存储器的分页管理

# ARM Linux

---

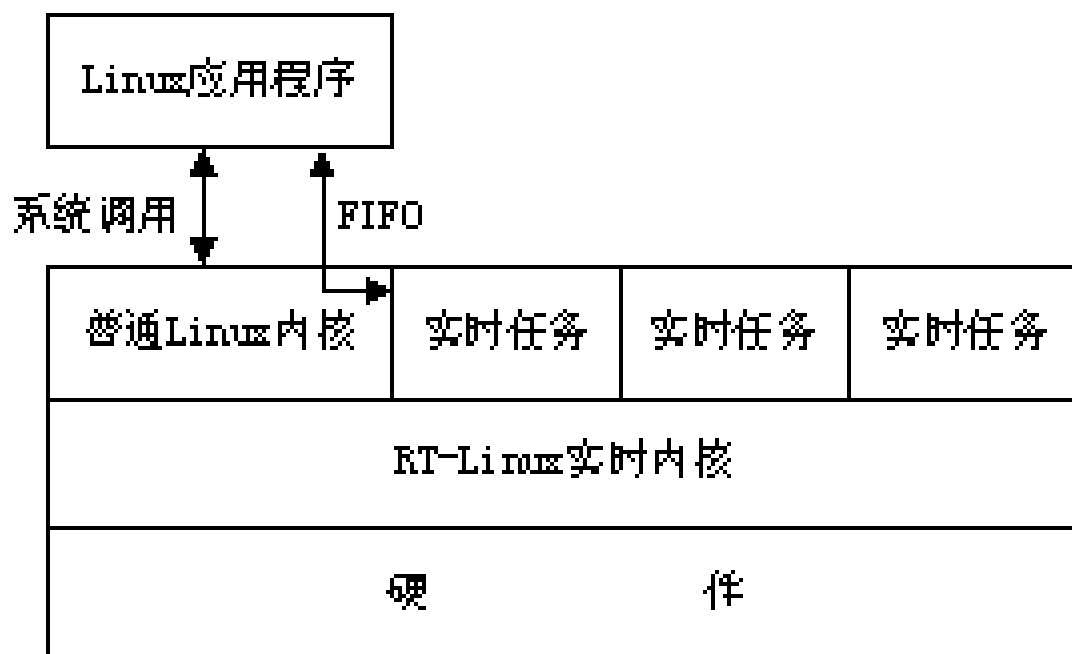
- } Linux标准发行内核中的一个分支
- } ARM Linux主要是由Russell King和其他人的共同努力，将Linux作业系统成功的移植在以ARM处理器为核心的机器上。
- } 与uClinux不同，支持带有MMU的ARM系列CPU  
如ARM610、ARM710、ARM720T、ARM920T、Intel StrongARM、Xscale等

## RT-Linux

---

- } 这是由美国墨西哥理工学院开发的嵌入式Linux操作系统  
RT-Linux开发者并没有针对实时操作系统的特性而重写Linux的内核，提出了精巧的实时核心，并把标准的Linux核心作为实时核心的一个进程，同用户的实时进程一起调度。这样对Linux内核的改动非常小，并且充分利用了Linux下现有的丰富的软件资源。
- } 到目前为止，RT-Linux已经成功地应用于航天飞机的空间数据采集、科学仪器测控和电影特技图像处理等广泛领域。
  - o

# RT-Linux的框架



# RT-Linux的关键技术

## } 减少中断封锁

通过软件来模拟硬件的中断控制器。当Linux系统要封锁CPU的中断时，RT-Linux中的实时子系统会截取到这个请求，把它记录下来，而实际上并不真正封锁硬件中断，这样就避免了由于封中断所造成的系统在一段时间没有响应的情况，从而提高了实时性。当有硬件中断到来时，RT-Linux截取该中断，并判断是否有实时子系统的中断例程来处理还是传递给普通的Linux内核进行处理。

## } 提高时钟频率

普通Linux系统中的最小定时精度由系统中的实时时钟的频率决定，一般Linux系统将该时钟设置为每秒来100个时钟中断，所以Linux系统中一般的定时精度为10ms，即时钟周期是10ms，而RT-Linux通过将系统的实时时钟设置为单次触发状态，可以提供十几个微秒级的调度粒度。

## } 意大利的RTAI

Real-Time Application Interface, 实时应用接口

源于RT-Linux, 它在设计思想上和RT-Linux完全相同。

} 它当初设计目的是为了了解决RT-Linux难于在不同Linux版本之间难于移植的问题

} 实时硬件抽象层, RTAI在Linux上定义了一个实时硬件抽象层 (RTHAL, Real Time Hardware Abstraction Layer), 实时任务通过这个抽象层提供的接口和Linux系统进行交互, 这样在给Linux内核中增加实时支持时可以尽可能少地修改Linux的内核源代码。

# Montavista

---

## } 1. 支持多种硬件及平台

支持6种CPU系列/22种CPU，其中包括IA-32/x86、PowerPC、StrongARM、XScale、SuperH、MIPS、ARM。

## } 2. 多种实用的开发工具包

其中包括交叉开发工具、C和C++语言工具、源代码调试工具和跟踪工具、目标配置工具、库优化工具、Java技术的支持,图形化Trace工具等等

## Montavista(2)

---

### } 3. 性能优越的实时性

(1) 完全的抢占性内核。(2) 透明的MontaVista Linux实时调度器。它具有以下优点：

- 提高Linux系统的响应速度；
- 提供标准的Linux API和以线程为基础的用户编程模式；
- 能够配置多种实时优先级；
- 能够处理毫秒级~150微秒级的调度需求。

MontaVista Linux实时调度器在Linux的调度器之前运行，然后由它来管理所有有实时要求的进程。如果不存在有实时要求的进程，那么MontaVista Linux实时调度器会十分迅速、有效地将控制权交给标准的Linux调度器。

## Wind River Linux

---

} 最新的Wind River Linux 3.0基于全新的Linux 2.6.27内核和GCC 4.3，提供了灵活、通用的开发环境。其中拥有超过500个增值软件包，较其前一版本产品增加了250多个，包括多媒体、图形处理和HMI(人机接口)技术等，例如X.org、GTK+、GNOME或GStreamer等。新版Linux平台还全面支持最尖端的多核处理功能，例如基于KVM(内核虚拟机)的可视化功能以及多核数据卸载(offload)等，使电子产品厂商能够更充分地发掘和利用多核硬件架构的潜能。

## Wind River Linux(2)

---

- } Wind River Linux 3.0还提供了面向垂直行业市场的预集成系统模板，典型的行业包括航空与国防、消费电子产品、工业自动化、网络通信和医疗设备等。在此次Wind River Linux 3.0发布前，风河Linux已针对各个垂直行业市场分别进行了程序包集成，实现了各行业所要求的特定功能特性和硬件支持。
- } Wind River Linux 3.0完全遵循Linux基金会制定的Carrier Grade Linux(运营级Linux)4.0规范，满足电信级和高端数据通信网络市场需求。能够实现全范围软/硬(soft-to-hard)实时性能特性。

# 诺基亚 Table OS



# 诺基亚 Table OS

---

} 开源网站的支撑

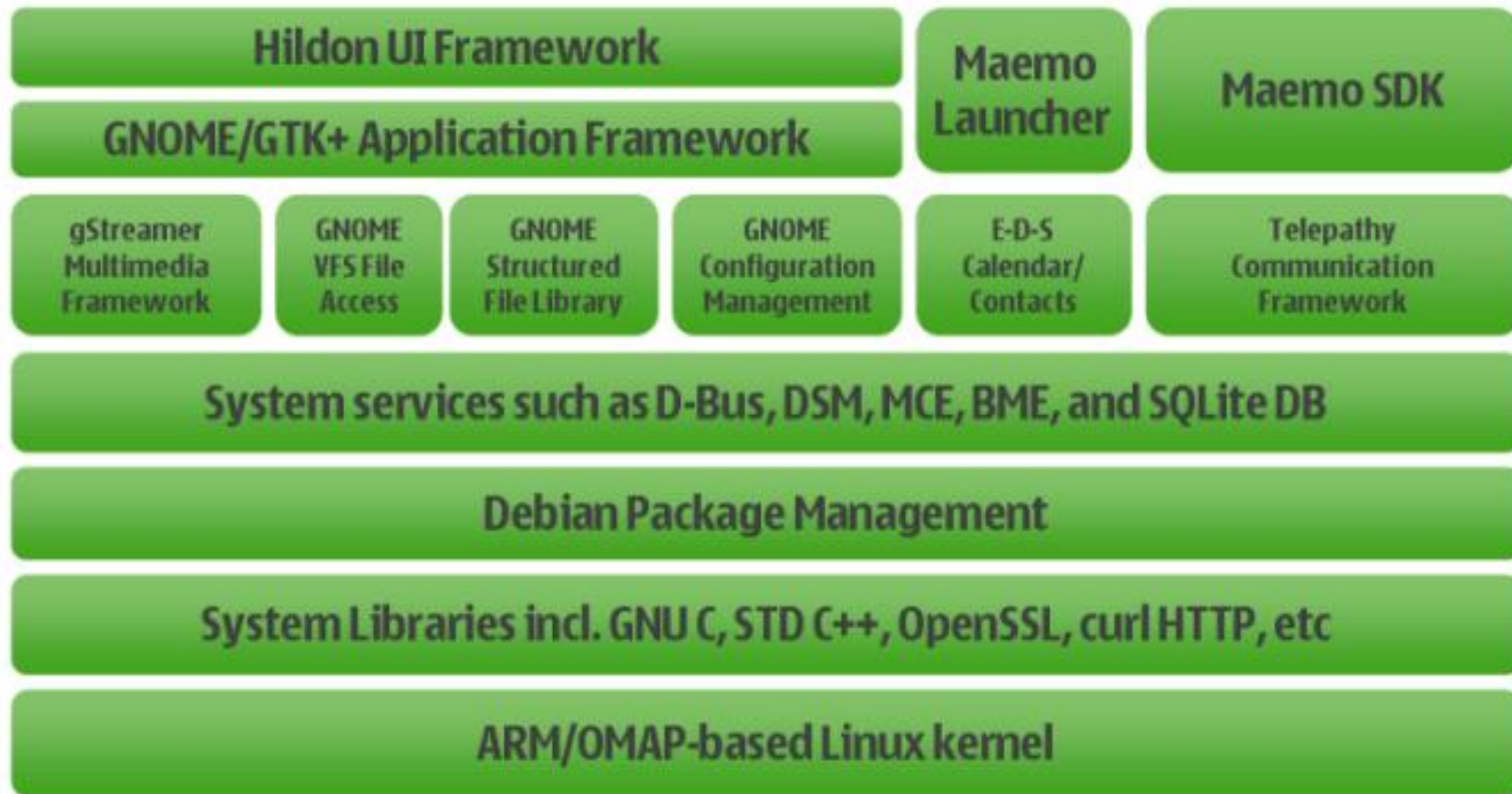
<http://maemo.org/intro/>

} 良好的maemo开发环境

} 基于Hildon UI架构

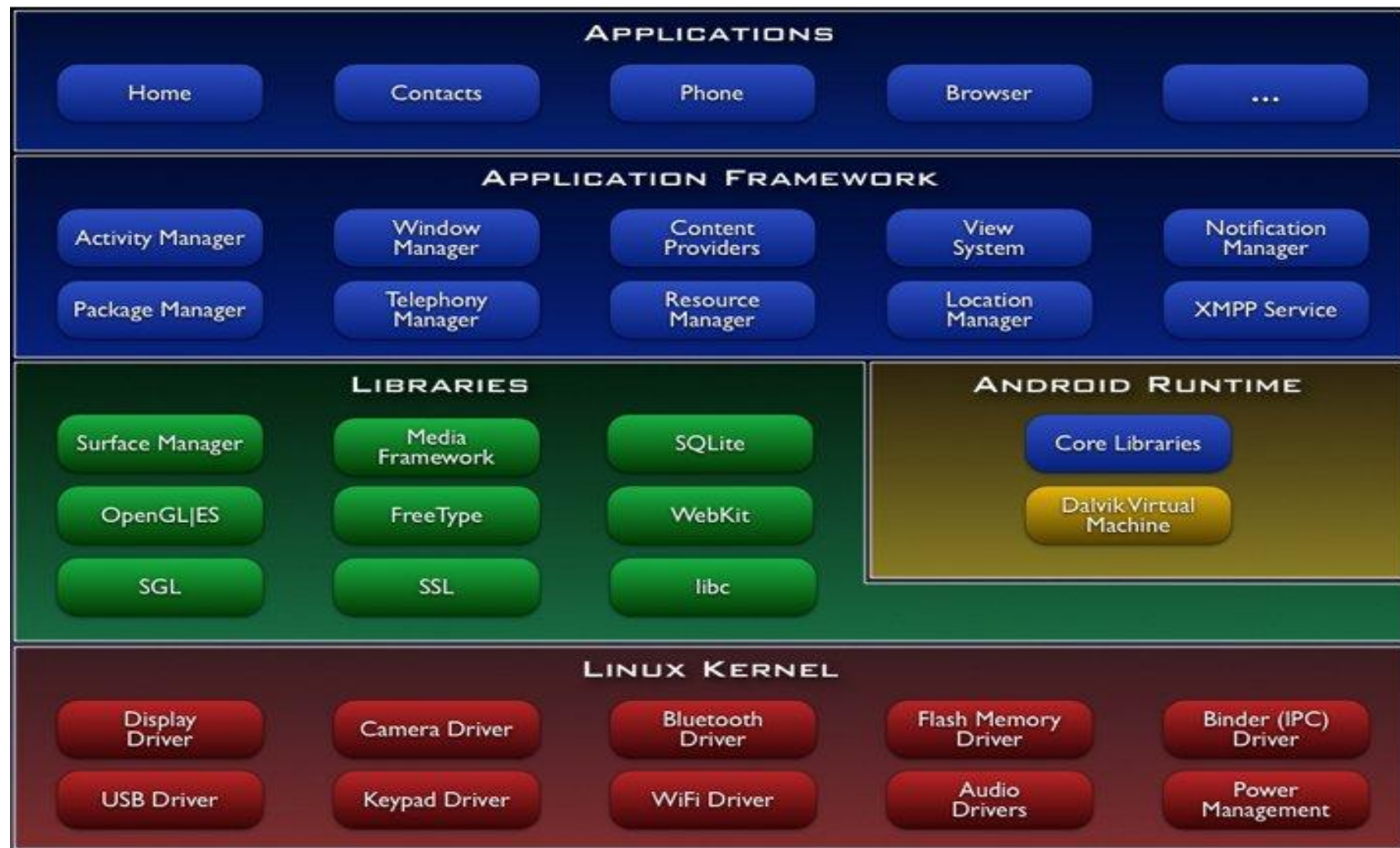
} 产品 N800, N900系列

# Maemo 软件架构



Android 是基于Linux平台的开源手机操作系统。它包括操作系统、用户界面和应用程序——移动电话工作所需的全部软件。Google与开放手机联盟合作开发了 Android，这个联盟由包括中国移动、摩托罗拉、高通、宏达电和 T-Mobile 在内的 30 多家技术和无线应用的领军企业组成。Google通过与运营商、设备制造商、开发商和其他有关各方结成深层次的合作伙伴关系，希望借助建立标准化、开放式的移动电话软件平台，在移动产业内形成一个开放式的生态系统。

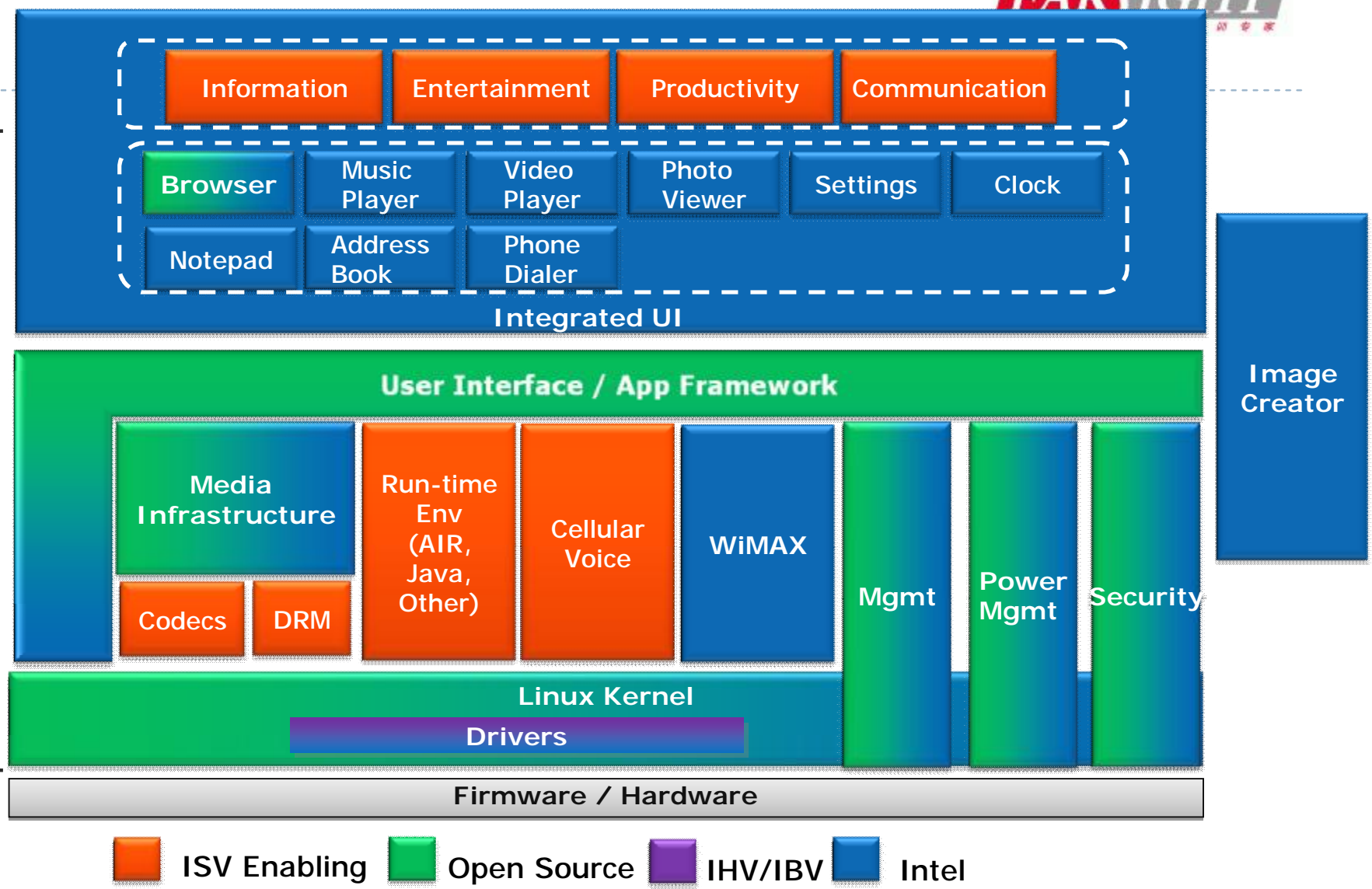
# Android 系统架构



## 英特尔Moblin

- } Moblin项目由Intel公司于2007年7月正式启动，最初主要集中于开发MID软件系统，基于Linux\* 的操作系统作为突破口出现了新的产品设备种类，能够访问Internet上丰富的内容，同时没有与PC相关的复杂性。这些设备从移动互联网设备（Mobile Internet Devices, MID）、车载信息系统、便携式媒体播放器（PMP）到个人导航设备（PND）、数码机顶盒（STB）到笔记本电脑等以各种形式推动新技术、新设备的发展。”
- } 基于Clutter构建的3D图形界面
- } 基于Atom低功耗IA架构
- } 支持Helix,gstreamer框架及各种媒体解码器

Integrated Linux Stack



## 驱动开发课程简介

---

- } 目标：没写过驱动的人可以独立编写驱动、写过驱动的人更上一层楼，成为驱动专家
- } 内容：
  - } Linux字符设备驱动程序
  - } 内核内存管理与中断处理
  - } 块设备驱动程序，MTD以及文件系统
  - } 网络驱动程序
  - } USB设备驱动程序开发
  - } 帧缓冲驱动程序和DMA
- } 嵌入式Linux设备驱动开发班（FSLD3001）

Thank You !