Evaluating Android OS for Embedded Real-Time Systems

Cláudio Maia, Luís Nogueira and Luís Miguel Pinho

OSPERT 2010, Brussels, Belgium
July, 06, 2010
Agenda

- Context
- What is Android?
- Dalvik Virtual Machine
- Linux Kernel
- Possible Directions
- Conclusion
Context

- Increasing relevance of Android in the mobile industry
- Open-source platform
  - Linux kernel based architecture
- Possibly a target to ERTS
- The paper presents the evaluation of the platform for ERTS with the objective of discussing potential ERTS support in Android
What is Android?

- Applications
- Application Framework
- Libraries
  - Android Runtime
    - Core Libraries
    - Dalvik VM
- Linux Kernel
Dalvik Virtual Machine

- Multiple independent processes, each one with its own address space and memory

- Implements the Java Concurrency Model
  - Shared Objects and Monitors

- Follows pthread model, separated by groups
  - VM Threads and Application Threads

- Garbage Collector presents the “Stop the world behavior”
  - Separated Instances that collect independently

- Scheduling and Interrupt Handling are delegated to the Linux Kernel
Linux Kernel

- Version 2.6.XX

- Core scheduler is periodic and preemptive based on scheduling classes' priority
  - Android uses the CFS by default

- Own implementation of libc – Bionic
  - Process-Shared Mutexes and Condition Variables are not supported
  - Thread Mutexes and Condition Variables are limitedly supported

- Interrupt Handling/Resource Management are not timely bounded
Possible Directions - I

Applications

Application Framework

Libraries

Android Runtime

Core Libraries

Dalvik VM

RT-JVM

Linux Real-Time

Evaluating Android OS for Embedded Real-Time Systems
Possible Directions - II

Evaluating Android OS for Embedded Real-Time Systems
Possible Directions - III

Diagram showing the layers of Android OS components from top to bottom:
- Applications
- Application Framework
- Libraries
  - Android Runtime
    - Core Libraries
    - Dalvik VM
- Real-Time Applications
- Linux Real-Time
Possible Directions - IV

Application Framework

Applications

Libraries

Android Runtime

Core Libraries

Dalvik VM

Real-Time Applications

Linux Kernel

Real-Time Hypervisor

Evaluating Android OS for Embedded Real-Time Systems
Chosen Direction & Challenges

- The first proposed direction is considered the one that causes less impact in the system as a whole
  - Android apps and QoS apps can coexist

- Scheduling operations at OS level
  - Handle each VM with the correct priority is a must

- Memory Management
  - Mechanisms to address system resource saving

- Synchronisation Mechanisms (If necessary)
  - Communication between each VM's threads
Conclusion

- Potential target for ERT environments
  - Industry would benefit from it

- The evaluation performed shows that determinism is not possible out of the box

- Each of the four directions enables the possibility of incorporating determinism into the platform in its own way

- Whichever the chosen direction, work must be conducted at different layers of the architecture
Questions?