

# Àndroit Real-time for the rest of us

Android Builders Summit 2012

Wolfgang Mauerer

Siemens AG, Corporate Competence Centre Embedded Linux  
Corporate Research and Technologies, *Open Source Platforms*



Copyright © 2012, Siemens AG. All rights reserved.

# Àndroit Real-time for the rest of us

Android Builders Summit 2012

Wolfgang Mauerer, Jan Sawallisch, Gernot Hillier

Siemens AG, Corporate Competence Centre Embedded Linux  
Corporate Research and Technologies, *Open Source Platforms*

S. Hönick, S. Oberthür, University Paderborn



Copyright © 2012, Siemens AG. All rights reserved.

# Overview

1 Introduction & Rationale

2 PREEMPT\_RT

3 Android Architecture

- Software Architecture
- Demonstration
- Performance Measurements

4 Building Your Own

# Outline

## 1 Introduction & Rationale

## 2 PREEMPT\_RT

## 3 Android Architecture

- Software Architecture
- Demonstration
- Performance Measurements

## 4 Building Your Own

# Android & Industrial Control I



## Android & Industrial Control II



## Android & Industrial Control II



# From Android™ to Àndroit

## From Android to Àndroit

- **Android** and real-time
- **À bon droit** – the right choice!

## Business

- Machine consolidation
- Multi-Core proliferation





# From Android™ to Àndroit

## From Android to Àndroit

- **Android** and real-time
- **À bon droit** – the right choice!

## Business

- Machine consolidation
- Multi-Core proliferation



# From Android™ to Àndroit

## From Android to Àndroit

- **Android** and real-time
- **À bon droit** – the right choice!

## Business

- *Developer availability*
- *Attractivity sells*



# From Android™ to Àndroit

## From Android to Àndroit

- **Android** and real-time
- **À bon droit** – the right choice!

## Business

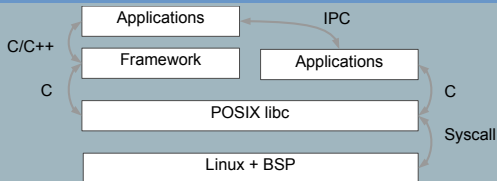
- *Developer availability*
- *Attractivity sells*

## Technical

- Reduced HMI complexity
- App model
- Opportunity: RT/non-RT separation
- Licensing

# Android $\neq$ Embedded Linux

## Architecture: (Embedded) Linux



## Key Concepts

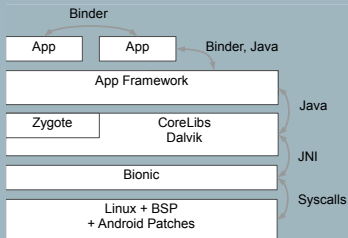
- Move complexity upwards
- Lingua Franca: Java
- Consistency

## On Philosophy

The whole is greater than the sum of its parts!

# Android $\neq$ Embedded Linux

## Architecture: (Embedded) Android



## Key Concepts

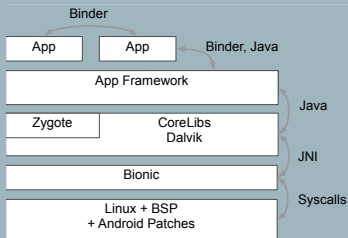
- Move complexity upwards
- Lingua Franca: Java
- Consistency

## On Philosophy

The whole is greater than the sum of its parts!

# Android $\neq$ Embedded Linux

## Architecture: (Embedded) Android



## Key Concepts

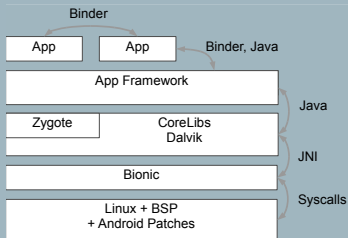
- Move complexity upwards
- Lingua Franca: Java
- Consistency

## On Philosophy

The whole is greater than the sum of its parts!

# Android $\neq$ Embedded Linux

## Architecture: (Embedded) Android



## Key Concepts

- Consistency for users *and* developers
- Convenient and safer languages

## On Philosophy

The whole is greater than the sum of its parts!

## Android: Use Cases

### Consumer

- In-car entertainment
- Home control

### Industrial

- Small PLCs
- Transportation and medical systems (Safety? Certification?)



# Outline

1 Introduction & Rationale

**2 PREEMPT\_RT**

3 Android Architecture

- Software Architecture
- Demonstration
- Performance Measurements

4 Building Your Own

# PREEMPT\_RT: Overview

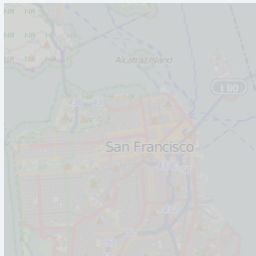
## Goal

Transform Linux into hard RT kernel

## Merge Strategy

Merge upstream, by end of year

## Roadmap



## Architecture



# PREEMPT\_RT: Overview

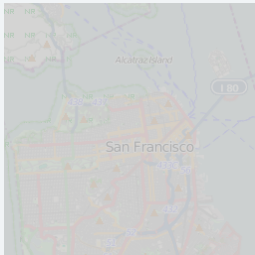
## Goal

Transform Linux into hard RT kernel

## Merge Strategy

Predictions: Learn from cold fusion

## Roadmap



## Architecture



# PREEMPT\_RT: Overview

## Goal

Transform Linux into hard RT kernel

## Merge Strategy

Predictions: Learn from cold fusion

## Roadmap



## Architecture



# PREEMPT\_RT: State of Affairs

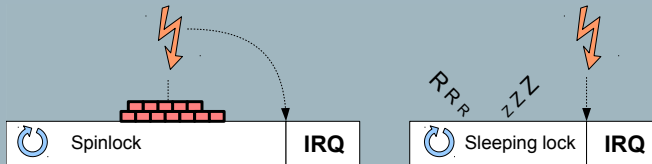
## Mainline

- High-res timers
- PI Mutexes
- Kernel preemption/Threaded interrupts
- Spinlock annotations

## Out of Tree

- Lock transformation
- Unconditional threaded interrupts
- Latency optimisation, bug fixes

## Illustration



# Outline

1 Introduction & Rationale

2 PREEMPT\_RT

**3** **Android Architecture**

- Software Architecture
- Demonstration
- Performance Measurements

4 Building Your Own

# Android Architecture: Challenges

## Challenges

- Android, preempt\_rt and BSP integration
- Propagating real-time “upwards”
- Consolidate RT and application model

## Asymmetry: HMI vs. RT



# Android Architecture: Challenges

## Challenges

- Android, preempt\_rt and BSP integration
- Propagating real-time “upwards”
- Consolidate RT and application model

## Asymmetry: HMI vs. RT

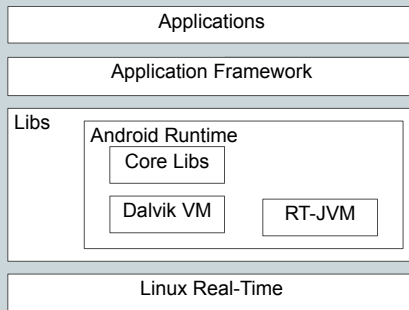




# Àndroit: Software Architecture I

## Architecture Variants

- Integrate RT-JVM
- Extend DVM
- Hybrid Approach



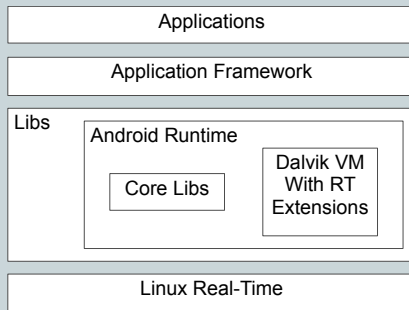
# Àndroit: Software Architecture I

## Architecture Variants

■ Integrate  
RT-JVM

■ Extend DVM

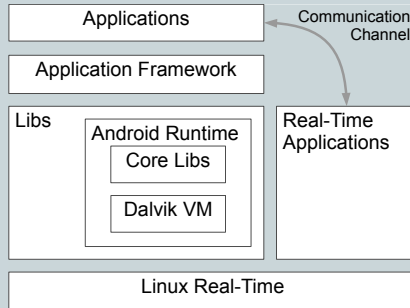
■ Hybrid Approach



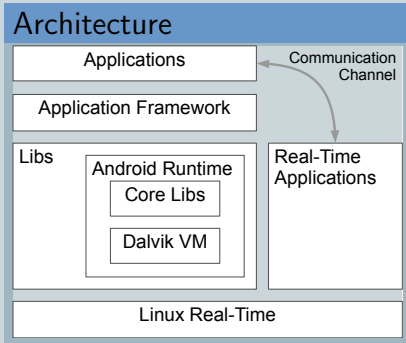
# Àndroit: Software Architecture I

## Architecture Variants

- Integrate RT-JVM
- Extend DVM
- Hybrid Approach



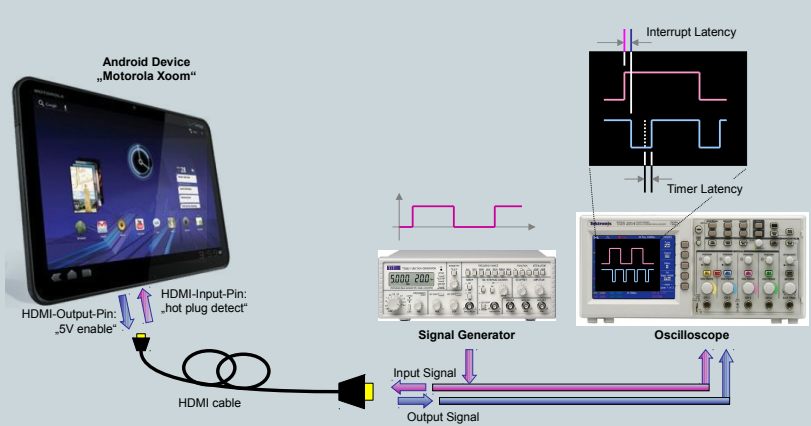
# Àndroit: Software Architecture II



## Design decisions

- Decouple RT and non-RT (Middleware?)
- Java coupling: RCU, transactions
- Introduce glibc

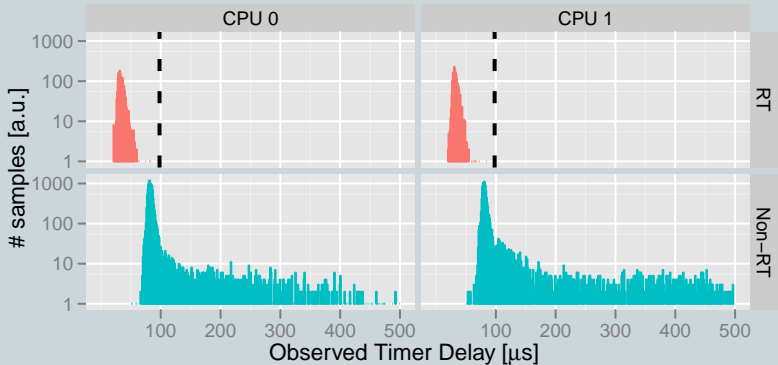
# Demonstration: Measurement Setup



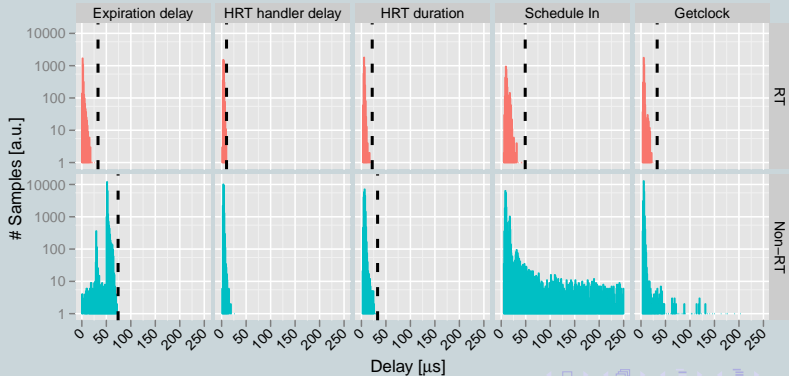
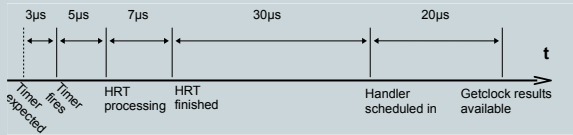
# Demonstration



# Performance Measurements I



# Performance Measurements I





# Outline

1 Introduction & Rationale

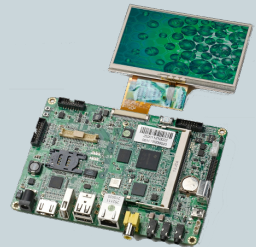
2 PREEMPT\_RT

3 Android Architecture

- Software Architecture
- Demonstration
- Performance Measurements

**4 Building Your Own**

## Good, Bad, Ugly



### Issues

- Device Access
- Sources/BSP
- Interfaces

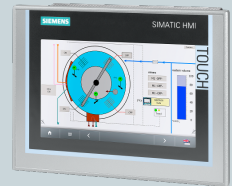
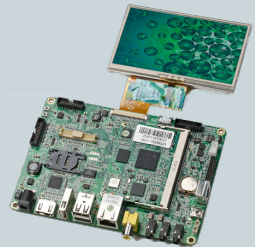


Image Sources: Motorola, gizmodo.de, Freescale, ww1.prweb.com, Wikipedia

# Good, Bad, Ugly



## BSP Theorem

$$R_{\text{BSP}}(n) \propto \frac{1}{n}$$

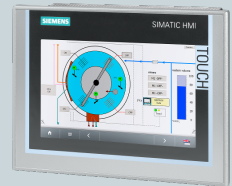
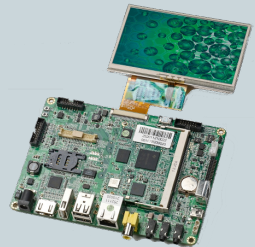


Image Sources: Motorola, gizmodo.de, Freescale, ww1.prweb.com, Wikipedia

# Good, Bad, Ugly



## Corollary

$$\lim_{n \rightarrow \infty} R_{BSP}(n) = 0$$

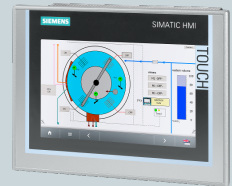
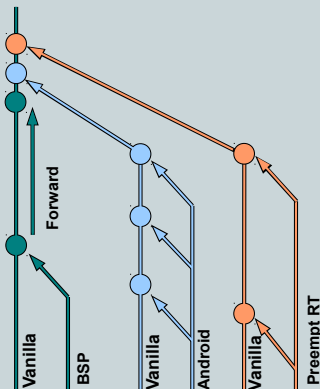


Image Sources: Motorola, gizmodo.de, Freescale, ww1.prweb.com, Wikipedia

# Android HOWTO



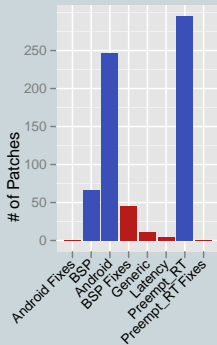
## Base System Tasks

- preempt\_rt-capable base kernel (BSP!)
- Android patches
- preempt\_rt patches
- Latency tuning (device drivers!)

## Userland Tasks

- Android component adaption
- RT control integration

# Android HOWTO



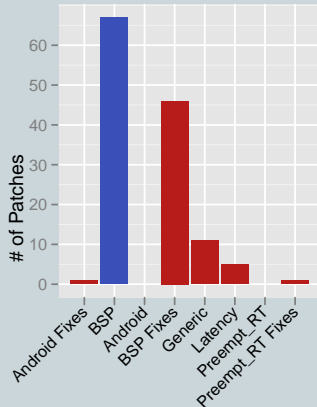
## Base System Tasks

- preempt\_rt-capable base kernel (BSP!)
- Android patches
- preempt\_rt patches
- Latency tuning (device drivers!)

## Userland Tasks

- Android component adaption
- RT control integration

# Android HOWTO



## Base System Tasks

- preempt\_rt-capable base kernel (BSP!)
- Android patches
- preempt\_rt patches
- Latency tuning (device drivers!)

## Userland Tasks

- Android component adaption
- RT control integration

# Thanks for your interest!