Real-Time Scheduling on L4-Based Microkernels

Suresh Iyer, Jaeheon Yi, Scott Brandt
June 1, 2007
L4-Embedded Microkernel

- State of the art microkernel
- High performance, IPC cost minimized
- Target: Small, fast, provably correct
- Platform for virtualization
- Industrial applications
Motivation

- L4 has priority-based scheduler
- Real Fast is not Real Time!
- Implementing RBED in L4 will allow:
  - Real-time processing
  - Resource isolation
RBED Scheduler

The RAD model

Rate-Based Earliest-Deadline
Integration of HRT / SRT / BE
Pseudo-periodic BE threads
Based on RAD model
Rate Allocation

HRT, 30% → 30% Usable → 60% Used → 10% BE Reserve

HRT welcome!
Rate Allocation

HRT, 40% → 10% BE Reserve
30% Usable → 60% Used

NO room for HRT
Rate Allocation

SRT, 40%

10% BE Reserve

30% Usable

60% Used

Can SRT fit?
Rate Allocation

- SRT, 30% allocated
- 30% Usable
- 10% BE Reserve
- 60% Used

We give only 30%
Rate Allocation

We give 10% plus extra among all 4 BE
Some Issues Encountered

Direct switch / Timeslice donation
High-accuracy timers
Coarse-grained allocation
Measurement accuracy of overhead
Results (1)

Utilization on L4
Results (2)

Dynamic Resource Allocation

![Graph showing dynamic resource allocation over time](image-url)
Results (3)

Scheduler Overhead Comparison
“A feature is only allowed in the kernel if this is required for the implementation of a secure system.”

Jochen Liedtke
Work in Progress

Resource virtualization?
  - Pools of threads concept

Support for legacy scheduling?
  - Migration path to rate-based scheduling
Thank you!

Questions?