

Realtime Audio vs. Linux 2.6

Lee Revell

Background

- History:
- LL patches for 2.4
- Beginning of 2.6 low latency work

Latency debugging mechanisms

- xrun debugging
- Ingo's latency tracer

BKL issues

- reiserfs 3.x was thought to be good for latency, however
- Its extensive use of the BKL (lock_kernel) caused latency problems, BUT
- Someone needs to retest it now that the BKL is preemptible

BKL issues, continued

- Virtual console switching was a big problem as the BKL was held
- This was resolved by making the BKL preemptible

Hardirq context

- IDE a big problem if LBA48 in use
- Solution: allow tuning the max request size via `/sysfs`
- ESSENTIAL to getting good latency with 2.6.x!

Process context

- Too many small problems here to list
- VFS, VM

Process context, continued

- Many issues in ext3 filesystem fixed
- Reservations, journalling, etc
- Choice of FS should have very little impact on latency these days

Softirq context

- Network softirqs
- Still a big problem – some cannot be worked around
- `rt_secret_rebuild()` is #1 latency problem with mainline now
- Proposals to fix it
- Or we could allow softirq threading

Performance issues

- Some latency problems were merely performance bugs
- kallsyms lookup used a very slow linear search with preemption disabled
- Fixed by improving the algorithm

Non-kernel factors

- Bad hardware/driver interactions (the VIA case)
- ACPI/SMM issues – still a big problem
- It's possible they can be worked around (RTAI code to disable them), but
- The best solution is still to avoid buying this hardware

The -rt kernel

- Hopefully most of you know what it is

Conclusions

- Between kernel 2.6.7 or so and 2.6.14, worst case latencies improved by almost an order of magnitude
- From 20-50ms to 2-5ms
- Further improvements are possible: route cache flushing, VM
- I believe we can do 1ms reliably without merging more of the -rt patch

Future developments

- Changes described above can achieve 1-2ms latency
- For better latencies (1ms and less) deeper changes to the kernel are needed
- Spinlock -> mutex conversions, IRQ threading
- Extend the domain of what is “RT safe”

Discussion/Q&A

- Questions?