Contribution ID: **107** Type: **not specified**

printk: Why is it so complicated?

Monday, 9 September 2019 12:45 (45 minutes)

The printk() function has a long history of issues and has undergone many iterations to improve performance and reliability. Yet it is still not an acceptable solution to reliably allow the kernel to send detailed information to the user. And these problems are even magnified when using a real-time system. So why is printk() so complicated and why are we having such a hard time finding a good solution?

This talk will briefly cover the history of printk() and why the recent major rework was necessary. It will go through the details of the rework and why we believe it solves many of the issues. And it will present the issues still not solved (such as fully synchronous console writing), why these issues are particularly complex and controversial, and review some of the proposed solutions for moving forward.

This talk may be of particular interest to developers with experience or interest in lockless ring buffers, memory barriers, and NMI-safe synchronization.

I agree to abide by the anti-harassment policy

Yes

Primary author: OGNESS, John (Linutronix GmbH)

Presenter: OGNESS, John (Linutronix GmbH)

Session Classification: LPC Refereed Track