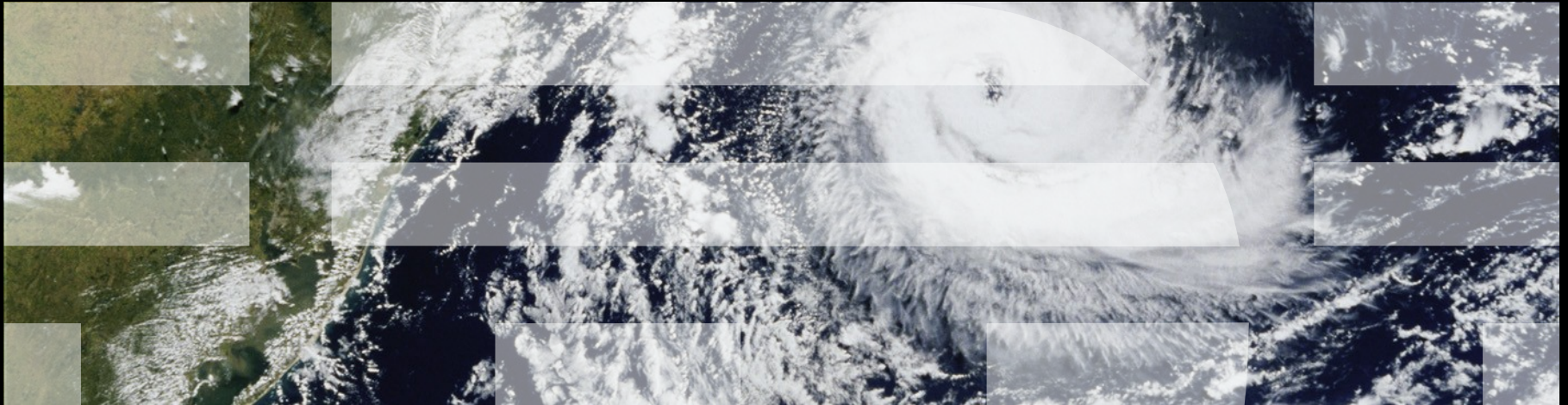




Frightening small children and disconcerting grown-ups: Concurrency in the Linux kernel

Jade Alglave (University College London); Luc Maranget (INRIA); Paul E. McKenney (IBM Corporation); Andrea Parri (Scuola Superiore Sant'Anna); Alan Stern (Harvard University)



What *Have* We Done? And Why???

First, Why???

- Linux kernel (LK) supports more than 30 CPU architectures
 - DEC Alpha, ARM, ARM64, IA64, MIPS, OpenRISC, PowerPC, RISC-V, S390, SPARC, x86, and 20 more
- Concurrent LK code must run correctly everywhere!!!
- Hence, “Interesting” memory-ordering discussions on the Linux kernel mailing list (Section 1 of paper)
 - How is core Linux-kernel code supposed to behave?
 - How must Linux-kernel synchronization primitives be implemented?
 - Does it behave correctly on exotic hardware?
 - What exactly can a Linux-kernel hacker get away with?*
 - And speaking as its main author, I am here to tell you that Documentation/memory-barriers.txt passed its sell-by date years ago...

So What Did We Do About All That???

Next, What???

- First formal memory model for the Linux kernel
 - Executable cat code, already used by Linux kernel hackers: Motivated removal of `spin_unlock_wait()` and DEC Alpha rework
 - Sections 2 and 3 of paper
- First memory model of any kind that includes RCU
 - Formulated fundamental law of RCU as well as the RCU axiom
 - Showed them to be equivalent (Section 4 of paper)
 - Verification tools can therefore use either fundamental law or RCU axiom
 - Showed that the userspace RCU library satisfies the fundamental law
 - (Section 6 of the paper)

So What???

Questions We Will Address During the Talk...

- Why not simply use the C++11 memory model???
- What do we do about out-of-thin-air accesses?
- The Linux kernel supports more than 30 CPU architectures, and is adding more (RISC-V). How does LKMM keep up?
- Why would anyone want *two* definitions of RCU???
- Exactly what use have Linux kernel hackers made of LKMM?
- If the Linux kernel memory model is so great, why hasn't it been accepted into the Linux kernel?

Questions We Will Address During the Talk...

- Why not simply use the C++11 memory model???
- What do we do about out-of-thin-air accesses?
- The Linux kernel supports more than 30 CPU architectures, and is adding more (RISC-V). How does LKMM keep up?
- Why would anyone want *two* definitions of RCU???
- Exactly what use have Linux kernel hackers made of LKMM?
- If the Linux kernel memory model is so great, why hasn't it been accepted into the Linux kernel?
- Most important: Who will be giving the talk? :-)

Legal Statement

- This work represents the view of the authors and does not necessarily represent the view of University College London, Microsoft Research, Inria-Paris, IBM, Oregon State University, Scuola Superiore Sant'Anna, or Harvard University
- IBM and IBM (logo) are trademarks or registered trademarks of International Business Machines Corporation in the United States and/or other countries.
- Linux is a registered trademark of Linus Torvalds.
- Other company, product, and service names may be trademarks or service marks of others.

NEXT!!!