

# Write once, herd everywhere

*Tuesday, 25 August 2020 07:00 (45 minutes)*

With Linux Kernel Memory Model introduced into kernel, litmus tests have been proven to be a powerful tool to analyze and design parallel code. More and more C litmus tests are written, some of which are merged into Linux mainline.

Actually the herd tool behind LKMM have models for most of mainstream architectures: litmus tests in asm code are supported. So in theory, we can verify a litmus test in different versions (C and asm code), and this will help us on 1) verifying the correct of LKMM and 2) test the implementation of parallel primitives in a particular architecture, by comparing the results of exploring the state spaces of different versions of litmus tests.

This topic will present some work to make it possible to translate between litmus tests (mostly C to asm code). The work provides an interface for architecture maintainers to provide their rules for the litmus translation, in this way, we can verify the consistency between LKMM and the implementation of parallel primitives, and this could also help new architectures to provide parallel primitives consistent with LKMM.

This topic will introduce the overview of the translation and hopefully some discussion will be made during or after the topic on the interface.

## I agree to abide by the anti-harassment policy

I agree

**Primary author:** FENG, Boqun

**Presenter:** FENG, Boqun

**Session Classification:** LPC Refereed Track

**Track Classification:** LPC Refereed Track (Closed)