

Memory management bits in arch/

Tuesday, 25 August 2020 10:00 (25 minutes)

Two significant parts of interaction between architectures and the generic MM are memory model (flat, discontigmem, sparsemem) and memory detection and initialization.

SPARSEMEM was designed as replacement for DISCONTIGMEM, but although sparse memory model was stable and robust for long time, there are still several architectures that require DISCONTIGMEM and the conversion is not as trivial as one might think. I'd like to discuss the trade-offs and challenges involved in this transition in a hope to remove the complexity associated with maintenance of both models.

While the necessity to support extra memory model translates into code complexity and maintenance burden, the lack of consistency in memory detection and initialization among the architecture may cause exposure of run time bugs. Moreover, absence of a generic abstraction for physical memory layout makes every architecture to reinvent the wheel and, for example, we have e820 with numa_meminfo on x86, memblock on ARM/ARM64 and memblock with device tree in PowerPC. I believe that reaching a consensus about a generic data structure that will describe physical memory layout, including bank extents, NUMA nodes span, availability of mirroring and hotplug, would be beneficial to all architectures.

I agree to abide by the anti-harassment policy

I agree

Primary author: RAPOPORT, Mike (IBM)

Presenter: RAPOPORT, Mike (IBM)

Session Classification: linux/arch/* MC

Track Classification: linux/arch/* MC