

CPU 是一个通用计算芯片, 对于一些在一定场景下要频繁进行的单一模式操作, 使用 CPU 既不高效, 也浪费 CPU 资源。

CPU 即是对特定并行计算进行处理

的芯片, CPU 只要下达任务给 CPU, 即可享受高速并行计算, 也能节省 CPU 通用计算处理资源。

同理, 对硬盘等相对于内存而言

的外设进行数据传输,尤其是非连续读写,每次都要经过I/O,且CPU进行数据读写操作,并不会用到计算能力,这是一种资源浪费。所以,单纯的数据搬运,更适合交给专门进行搬运的芯片来处理,这就是DMA。

Kafka 是针对数据搬运进行优化的通道。有了DMA之后,cpu

也会自动将外设或硬盘与内存之间的数据搬运交给 DMAc，但对于数据在内存中的流通，用不到 DMAc，cpu 也没有对其进行优化。以 web 数据发送为例，kafka 即通过优化数据流通，减少数据在内存中的转移操作，以零拷贝如快速度。

数据发送给 web 是来自用户态的

服务器应用的需求，cpu按照常规流程，要将从硬盘复制到内存（这个命令由内核态执行），得到的数据存于内核态区，再由内部socket号发给请求应用，再由应用将数据发给网络卡，要经过。整个过程经过了两次由内核态到用户区的拷贝，这是没必要的，但按照应用与cpu的沟通，这是流程。

... 5.2.2 右左左需要

kafka 即优化{程序, 且通过 100% 带宽带上, 在数据从硬盘进入内存后, 直接发送到网卡。这样, 数据在内存中没有拷贝过, 全部拷贝都由 DMA 处理, 即节省 CPU, 又加快效率。

注意, DMA 只是为 CPU 减负, 它处理能力并不比 CPU 快; 而 kafka 也不是因为用了 DMA 才快, 而是减

... .. DMA 并不因

少了搬运次数所以快。不仅不用

DMAC，只要kafka是零拷贝，就会

更快，零拷贝才是快的原因。