



(/)

技术 (/Vppproject/Vpptech)
安装 (/Vppproject/Vppinstalling/
建造 (/Vppproject/Vppbuilding/
特征 (/Vppproject/Vppfeatures/)

向量封包处理器 (VPP)

VPP如何运作

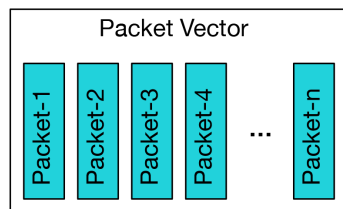
FD.io背后的技术

FD.io的核心是矢量数据包处理 (VPP) 。

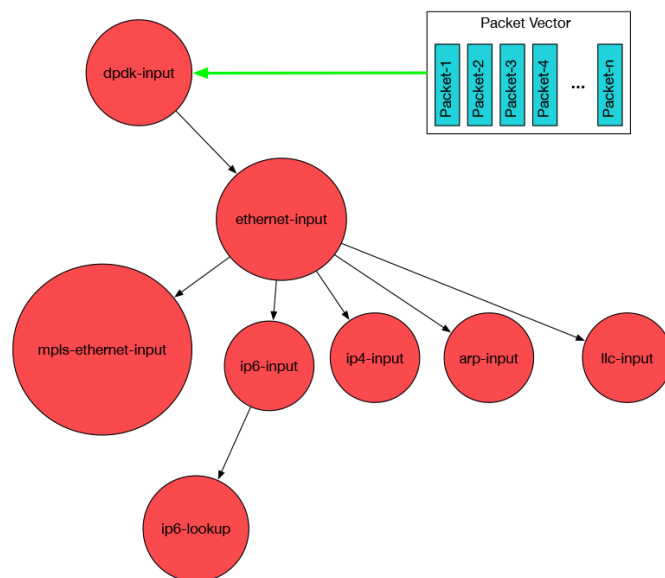
从2002年开始开发，VPP是当前在运输产品中运行的生产代码。它在x86服务器和嵌入式设备上的x86，ARM和Power架构等多种架构的用户空间中运行。VPP的设计与硬件，内核和部署（裸机，VM，容器）无关。它完全在用户空间中运行。

VPP帮助FD.io突破了性能和规模的极限。独立测试表明，以VPP为基础的FD.io在规模上比当前可用的技术快两个数量级。

VPP从网络IO层读取最大的可用数据包向量。



然后，VPP通过“数据包处理”图处理数据包的向量。



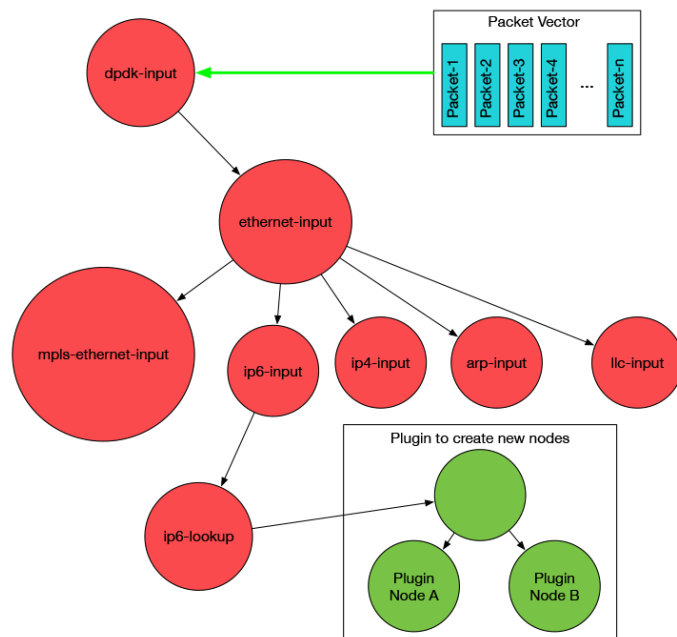
VPP不会处理整个图形中的第一个数据包，然后处理整个图形中的第二个数据包，而是先处理通过一个图形节点的所有数据包向量，然后再移动到下一个图形节点。

由于向量中的第一个数据包会预热指令缓存，因此其余数据包往往会以极高的性能进行处理。处理数据包向量的固定成本在整个向量中摊销。这不仅导致非常高的性能，而且还导致统计上可靠的性能。如果VPP落后一点，则下一个向量包含更多的数据包，因此固定成本将在更大数量的数据包中摊销，从而降低了每

个数据包的平均处理成本，从而导致系统追赶。结果，吞吐量和等待时间非常稳定。如果有多个核可用，则图形调度程序可以将（向量，图形节点）对调度到不同的核。

粗略的日常比喻是要考虑一堆木材的问题，其中每一块木材都需要切割，打磨并在上面钻个孔。一次只能拥有一个工具（类似于指令缓存）。如果您先拿起锯子并进行所有切割，然后拿起砂光机并进行所有打磨，然后拿起钻头并进行所有操作，则将更快地完成木材的切割，打磨和钻孔。您的钻探。拿起每个木材的工具要慢得多。

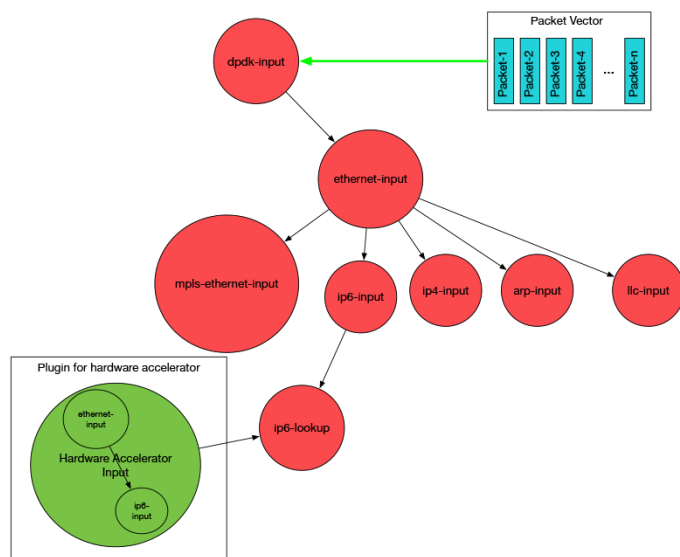
VPP的图形节点体系结构还易于扩展。您可以从单独的源代码库中为VPP构建独立的二进制插件（只需要标头）。插件从插件目录加载。VPP插件可以重新排列数据包图并引入新的图节点。这允许通过插件引入新功能，而无需更改核心基础结构代码。



硬件加速

这种相同的图节点体系结构还允许FD.io在可用时动态地利用硬件加速，从而使供应商能够继续在硬件方面进行创新，而不会违反FD.io软件的“随处运行”的承诺。

从逻辑上讲，您可以将硬件加速视为将某些软件图形节点替换为执行相同功能的硬件，只是速度更快。因为硬件加速器供应商可以通过简单地提供一个充当输入节点的插件来实现此目的，该插件可以移交给要在软件中执行功能的第一个图形节点，或者提供一个输出节点就可以尽快移交该插件。软件处理完成。这样，可以使用加速硬件（如果存在），但是如果功能缺失或资源用尽，则功能可以继续。这种架构为硬件加速器提供了最大的创新空间。



VPP入门 (/vppproject/vppinstalling)

©2019快速数据项目。版权所有©2018 FD.io Project, 一系列LF Projects, LLC

有关网站使用条款，商标政策和其他项目政策的信息，请参阅 LF Projects (<https://lfpjects.org/>)。

领空雨果主题 (<https://themes.gohugo.io/airspace-hugo/>) 版权©由Themefisher (<https://themefisher.com/>)设计与开发。 知识共享署名3.0许可。 (<https://themefisher.com/license/>)