

Close to the edge

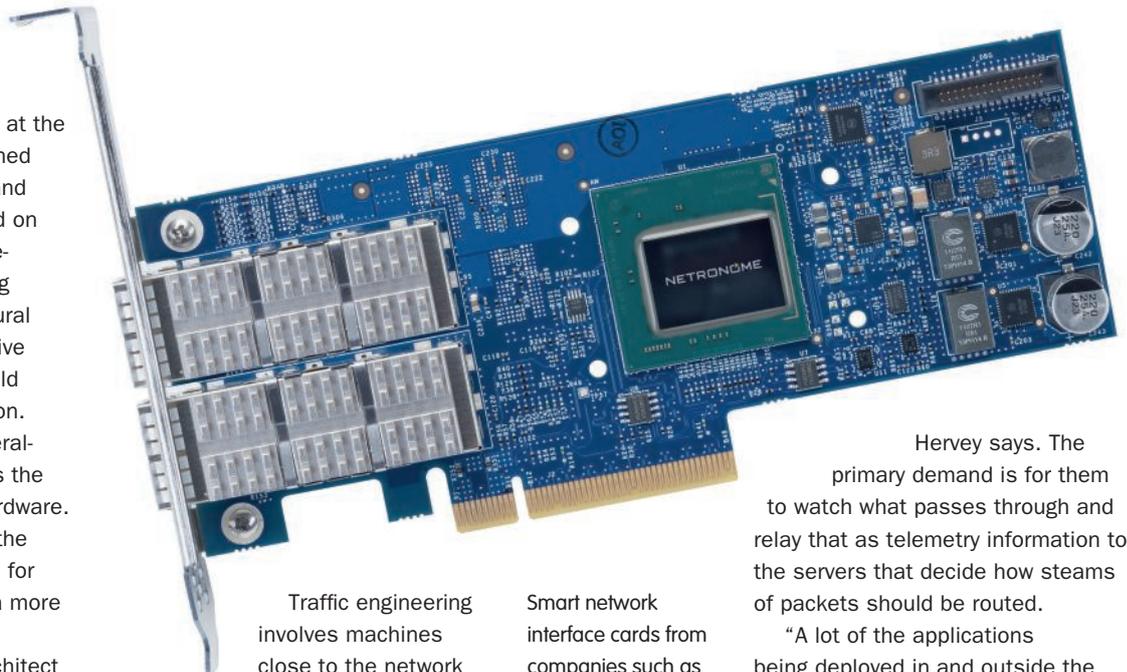
Software control is pushing network processors further to the edge. By **Chris Edwards**.

There is a tension that lies at the heart of the software defined network (SDN). Telecom and datacom operators have seized on it because the idea of software-based control makes everything more flexible and it is only natural that demand for more responsive communications services should push technology in that direction. But software that runs on general-purpose microprocessors lacks the efficiency of carefully tuned hardware. That is a major problem when the customers' other demands are for networks simply to deliver data more quickly and cheaply.

George Hervey, principal architect at Marvell, says one of the biggest driving forces for network capacity is within and between data centres. The workloads they run make heavy use of virtual machines that hop from server to server, based on available capacity, and call on specialised accelerators to handle data analysis or machine learning jobs.

"We see a tremendous growth in distributed compute nodes that need to share information with each other through the network," Hervey says. "People today think of networks being driven by more subscribers. The component of the growth that is usually left out is machine to machine. Distributed workloads are causing the growth. People don't understand how big this is."

"The network is going to be all about speeds and feeds. The only way to handle this and not be overwhelmed completely is to have smart traffic engineering."



Traffic engineering involves machines close to the network but not necessarily the switches themselves making high-level decisions about where streams of packets flow. Video from a server may take one route, while control packets take another that offers lower latency, but cannot handle high bandwidths. In many cases, this is achieved by setting up virtual overlay networks using protocols such as VXLAN or tunnels. In response, core network switches become more streamlined and make fewer policy-based decisions. "The networks themselves don't need [microprocessor-level] intelligence for forwarding," Hervey contends. "Network processors just aren't able to cope with the speeds and feeds. People are willing to give up the traditional value of network processors to offer more bandwidth and connectivity."

Forwarding engines themselves do more than simple table lookups,

Smart network interface cards from companies such as Netronome have converged on an architecture that is, effectively, a smarter network processor

Hervey says. The primary demand is for them to watch what passes through and relay that as telemetry information to the servers that decide how steams of packets should be routed.

"A lot of the applications being deployed in and outside the network are not being deployed in a traditional way. Operators need telemetry to understand how they are being deployed," Hervey says. "There is a tremendous desire for knowledge and diagnostic ability. People want to know what's going on in their network."

Although they are now armed with more telemetry and other information from the network, the processors at the endpoints are struggling with the workload. This is leading to a further change in the balancing act between packet forwarders and processors. At the August 2017 Future:Net conference, organised by virtual machine specialist VMware, Mellanox' vice president of software architecture Dror Goldenberg explained the problem. A 200Gbit/s Ethernet switch needs to be able to parse, sort and forward some 16million maximum-sized frames a second. With much smaller 64byte

packets, that number balloons to almost 300m. With the larger packets, a processor in a software-only switch has just 65ns to deal with each one – enough time for five level-three cache accesses or a call to the operating system. “With 64byte packets, you basically have budget for nothing,” he argues.

With today’s multicore processors, the ability to spread the workload out gives each processor core more time to do its job. But it imposes a heavy burden on servers expected to deal with the workload. Goldenberg says a software packet handler, such as Open vSwitch that runs on processors such as Intel Xeons, can consume half of the 16 cores of a high-end processor handling traffic at 10Gbit/s. “When datarates go from 10 to 25 or 40 to 100Gbit/s, the problem gets exacerbated. You end up losing almost all these cores,” he explains. In a data centre, ‘you’ve lost half of your compute, which is not acceptable’.

The answer being pursued by a number of silicon suppliers is to build smart network interface cards (smart-NICs) that can move workload back off the server processors. Although smart-NICs can be based on dedicated switch chips and FPGAs, suppliers ranging from Broadcom to

Netronome have converged on an architecture that is, effectively, a smarter network processor. It has multicore processors that can run Linux, combined with a selection of accelerators for functions such as packet encryption and content-addressable memories to quickly find destination addresses.

“I believe network virtualisation is a killer app for smart-NICs, especially when you go to data rates beyond 10Gbit/s,” says Sujal Das, chief strategy and marketing officer at Netronome.

The biggest problem facing smart-NIC builders is how they interact with the rest of the server. Only in a few cases will the smart-NIC offload all network processing from the Xeons on the main server blade. Some traffic will pass straight through the smart-NIC, but some packets need to be steered into the server for more extensive processing. This has subtle effects felt throughout the network.

Dejan Leskaroski, senior product manager for network virtualisation software specialist Affirmed Networks, explained at the OPNFV Summit in June that pushing some processing in smart-NICs makes it harder for the high-level services to see billing information. “We don’t see all the flows. And, with firewalls,



“I believe network virtualisation is a killer app for smart-NICs, especially when you go to data rates beyond 10Gbit/s.”

Sujal Das

how can a virtualised network function enforce policies when flows are bypassing it?”

The combination of software and hardware processing in smart-NICs brings problems of standardisation. The first wave of network function virtualisation software, based on projects such as ARM’s OpenDataPlane or Intel’s DPDK, focused on software that would be compiled onto CPUs. Those functions need to be translated into a mixture of microcode and register settings for network accelerators running alongside binaries for a Linux-compatible processor. Although server vendors and users could simply decide to standardise on a single vendor’s product line, this is not a popular option.

Broadcom distinguished engineer Fazil Osman believes: “When we talk to server vendors and ask what would make it easy for them to put in their systems, they want something that looks like a network device or storage device. The application doesn’t see it: it just sees it as an Ethernet device or a block device. The rest is done by an ecosystem of developers.”

The industry is struggling to find a standard that works across all forms of hardware. One possibility is to use a specialised language, such as P4 – a declarative language in the same vein as SQL, but for packet handling. Another is to use LLVM compiler technology to take applications written in a general-purpose language, such as C, and translate them on the target into binaries that can call on programmable hardware accelerators.

Das says: “I call it the Holy Grail of programming. You write code once and you don’t know where it’s running, it just runs in the right place.”

Goldenberg promises: “We will take care of the underlying plumbing.”

