NVIDIA’s Resource Transmutable
Network Processing ASIC
Kevin Deierling, VP Networking | August 29, 2023
From Programmability to Transmutability

- Decline of Moore’s law → Need for domain-specific architectures
- Goal → Hardware as flexible as software

**Current focus on programmability**
- Flexibility to perform a wide range of tasks
- Portability where possible

**Future focus on resource transmutability**
- Dynamic reprogramming of tasks
- Fungible resource allocation
Existing Challenges

- Traditional programmable ASICs: Fixed functions are limited in-runtime modification
- Current process: Risky, complex, not agile
  - Network level: Drain network flows and rerouting traffic, update, then bring back online
  - Device level: Prepare new program in scratch area, then switch over when complete
- Comparison to software data planes where:
  - Upgrades are straightforward
  - New functionality is easy to deploy
  - Programmability is flexible
  - Resource allocation is fungible

Conclusion — Transmutability is a must
Dynamic Workloads Require Transmutability

- Generative AI and Real-time AI cybersecurity frameworks are dynamic and evolving
  - Generative LLM AI and retrieval augmented generation
  - Real-time Mitigation: Precise threat response by injecting mitigation modules.
  - Monitoring of traffic patterns and digital fingerprinting of devices, users, and machines
  - Smart telemetry/filtering/sampling and real-time deep data analytics allows GPU to detect anomalous or divergent behavior
  - Dynamic automated quarantining, deep packet inspection, mitigation and restoration

- Just-in-time Network Optimizations: Quick detection, incorporation, and removal of policy
- Scenario-specific Network Extensions: Direct tenant program extensions and integrations
NVIDIA’s Solution: Transmutable ASICs

- Based on NVIDIA’s BlueField and Spectrum network ASICs
  - Dynamic resource allocation & reclamation
- Reprogram without packet drops, no down time
  - Low level primitives “add”, “remove”, “update”
  - Indirection - tables referenced by HW “pointers”
  - Full resource utilization - shared memory across all HW match-action processing units
- NVIDIA software stack + runtime changes ⇒ transmutable
  - *BlueField DPU*: NVIDIA P4, DOCA Flow, DPDK
  - *Spectrum Switch*: NVIDIA P4, SAI, Switch SDK
- Programmable throughout deployment with a new set of control plane APIs
  - P4Runtime extensions, backwards compatible
  - DOCA APIs
NVIDIA’s Disaggregated Architecture

**Reconfigurable Match-Action Tables (RMT)**
- Programmable pipeline architecture for packet processing
- Apply action “instructions” to a packet by matching keywords in the packet header vector
- Match can be exact, ternary, range or longest prefix match (LPM)

**NVIDIA’s Enhanced Disaggregated RMT (dRMT)**
- Compute and memory are disaggregated
- Shared memory is sharded, and accesses are load-balanced
- Match-action processors handle packets in parallel with run-to-completion model
- Enables granular reconfiguration and transmutability

---

**RMT architecture**

**dRMT architecture**
DPU Transmutable Pipeline SDKs

Transmutable Pipeline
- Runtime loadable
- Hybrid Pipelines
- Plug-n-Play

NVIDIA P4
- High level packet processing programming language
- Domain Specific compiler + open source P4Runtime API

DOCA Flow
- High level accelerated networking pipeline API

DPDK
- Low level polled packet processing API
ASIC Design and Architecture Features

- **Disaggregated Architecture** → Breaks resource allocation boundaries for partial reconfiguration
- **Sharded Resource Allocation** → Balances loads, avoids contention
- **Hybrid Programmability** → Efficient fixed modules + customization
- **Indirection** → Low-latency, efficient reconfigurations
- **Extended Control Plane** → Modify elements, 3 consistency guarantees

![Diagram showing program, element, and execution consistency with weaker consistency and lower transient overhead.](image-url)
Real-World Use Cases

- Benchmarks performed on NVIDIA Bluefield DPU and Spectrum switch
- Demonstrated scalability and adaptivity
- Server Load Balancer (SLB)
  - Perform optimizations at runtime to maximize throughput
- Source Based Routing and Telemetry
  - Pipeline extensions and chaining of P4 services
  - Dynamically extend pipeline with new functionality
  - Temporarily add in-situ network visibility
Server Load Balancer on BlueField

- “Pipeleon” runtime monitoring of rules/entries
  - High insertion rate event causes the cache table to “miss”
  - Miss counter threshold triggers a dynamic table reordering → throughput returns to line rate
- “Pipeleon” runtime monitoring of traffic and drops
  - Traffic pattern changes, causing a large number of policy driven packet drops
  - Drop counter threshold triggers a dynamic table reordering → throughput returns to line rate
"ELMO" source routed multicast
a. Enhancement to standard switch multicast table management
b. Encodes multicast group information inside packets → scale improvement

Postcard telemetry
a. Dynamically load a pipeline module to send telemetry data
b. Dynamically remove module once visibility no longer required

Accelerated Multicast on Spectrum
Conclusion & Next Steps

• NVIDIA’s innovation enables a truly adaptive network core, enabling network processing with resource transmutability
• Bridging the gap between hardware and software
• Transmutability as the future of network ASIC design
• Roadmap
  • Design the right APIs needed to load, control, update transmutable pipelines
  • Consistency guarantees and atomicity requirements
  • End to end solutions across multiple programmable network devices
  • Provide frameworks for performance and flexibility, but also complexity and scale
References

**Runtime Programmable Switches**
Jiarong Xing, Kuo-Feng Hsu, Matty Kadosh, Alan Lo, Yonatan Piasetzky, Arvind Krishnamurthy, and Ang Chen  
*NSDI 2022*  
https://www.usenix.org/conference/nsdi22/presentation/xing

**Unleashing SmartNIC Packet Processing Performance in P4**
Jiarong Xing, Yiming Qiu, Kuo-Feng Hsu, Songyuan Sui, Khalid Manaa, Omer Shabtai, Yonatan Piasetzky, Matty Kadosh, Arvind Krishnamurthy, T. S. Eugene Ng, Ang Chen  
*ACM SIGCOMM’23, New York, NY, September 2023*  

**A Vision for Runtime Programmable Networks**
Jiarong Xing, Yiming Qiu, Kuo-Feng Hsu, Hongyi Liu, Matty Kadosh, Alan Lo, Aditya Akella, Thomas Anderson, Arvind Krishnamurthy, T. S. Eugene Ng, and Ang Chen  
*HotNets’21*  

**Elmo: Source Routed Multicast for Public Clouds**
M. Shahbaz, L. Suresh, J. Rexford, N. Feamster, O. Rottenstreich and M. Hira  
*IEEE/ACM Transactions on Networking, vol. 28, no. 6, Dec. 2020*  

**Realizing Source Routed Multicast Using Mellanox’s Programmable Hardware Switches**
Matty Kadosh, Yonatan Piasetzky, Barak Gafni, Lalith Suresh, Muhammad Shahbaz, Sujata Banerjee  