Infinity: A Scalable Infrastructure for In-Network Applications

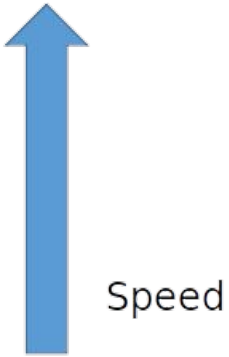
Marcelo Abranches (University of Colorado)
Karl Olson (University of Colorado)
Eric Keller (University of Colorado)

Needs for new generation Internet architecture

- New applications
- Legacy applications
- Tactile Internet
  - Ultra-low latency
  - High-availability
  - Reliability
  - Security

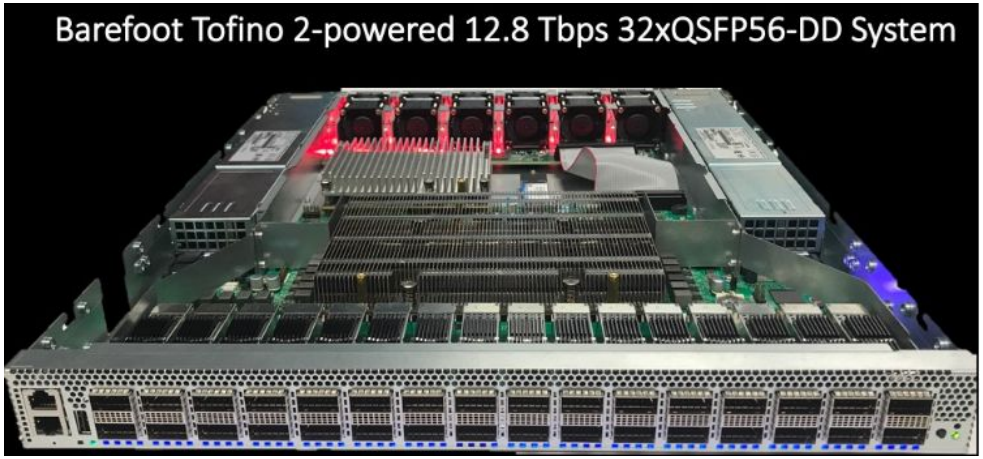
Source:https://www.attsavings.com/resources/internet/speed-guide/what-is-high-speed-internet/

Challenges with the Current Infrastructure



- Network speed increases at much faster pace than CPU speeds
- End of Moore’s Law
- Need to process more data with low latency to address real time demands
- We need more in-network processing

Programmable Networking Hardware



- Programmable ASICs
- Line-rate processing (up to 12 Tbps)
- High-performance
- Efficient use of data center infrastructure
- 1 P4 LB can replace ~277 software LBs (SilkRoad - Sigcomm ‘17)

In-network Computing

SilkRoad: Making Stateful Layer-4 Load Balancing Fast and Cheap Using Switching ASICs

Rui Miao, University of Southern California
Hongyi Zeng, Facebook
Changhoon Kim, Barefoot Networks
Jeongkeun Lee, Barefoot Networks
Minlan Yu, Yale University

NetCache: Balancing Key-Value Stores with Fast In-Network Caching

Xin Jin[1], Xiaozhou Li[2], Haoyu Zhang[3], Robert Soulé[2,4], Jeongkeun Lee[2], Nate Foster[2,5], Changhoon Kim[2], Ion Stoica[6]
[1]Johns Hopkins University, [2]Barefoot Networks, [3]Princeton University, [4]Università della Svizzera italiana, [5]Cornell University, [6] UC Berkeley

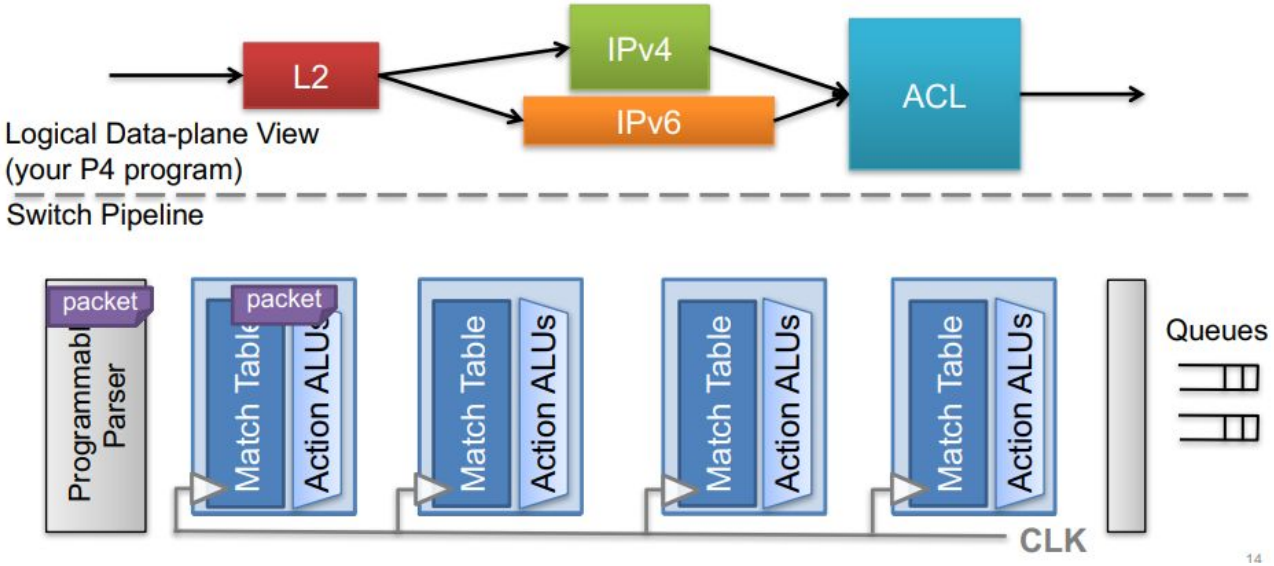
Scaling Distributed Machine Learning with In-Network Aggregation

Amedeo Sapio*, KAUST
Marco Canini*, KAUST
Chen-Yu Ho, KAUST
Jacob Nelson, Microsoft
Panos Kalnis, KAUST
Changhoon Kim, Barefoot Networks
Arvind Krishnamurthy, University of Washington
Masoud Moshref, Barefoot Networks
Dan R. K. Ports, Microsoft
Peter Richtárik, KAUST

NetPaxos: Consensus at Network Speed

Huynh Tu Dang* Daniele Sciascia*
Marco Canini† Fernando Pedone* Robert Soulé*
*Università della Svizzera italiana †Université catholique de Louvain

Challenges with Programmable Networking Hardware

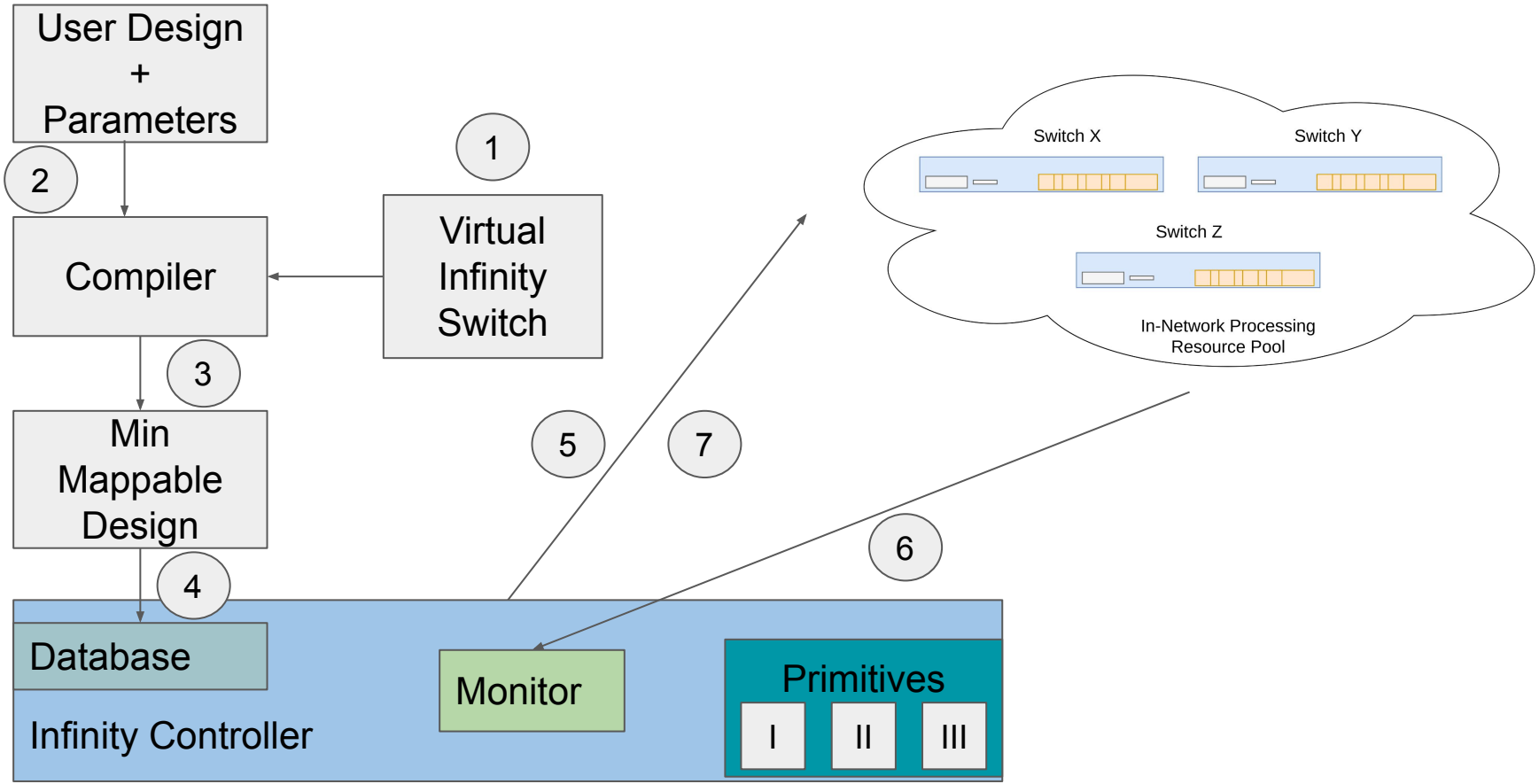


- Limited number of processing stages
- Limited amount of fast memory

Introducing Infinity

- In-network apps have the illusion of having infinite resources (processing stages and memory)
- More flexibility in deploying and sharing resources among in-network apps
- Primitives that enable aggregating distinct P4 switches resources into virtual aggregated hardware sets

Infinity Overview



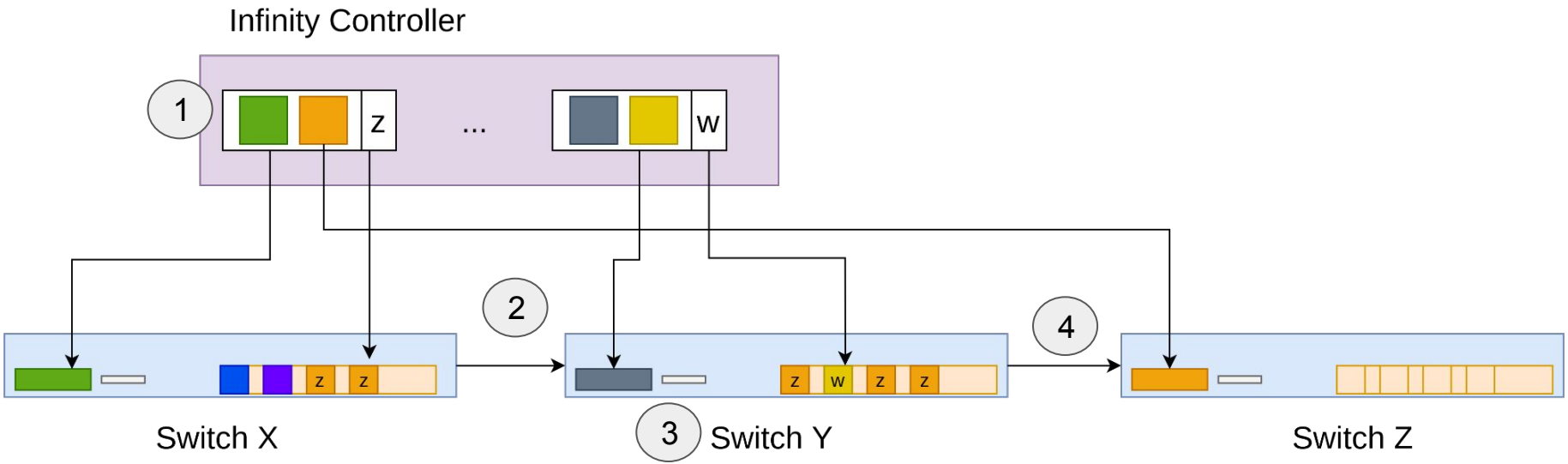
Infinity Primitives

Break processing pipelines into segments

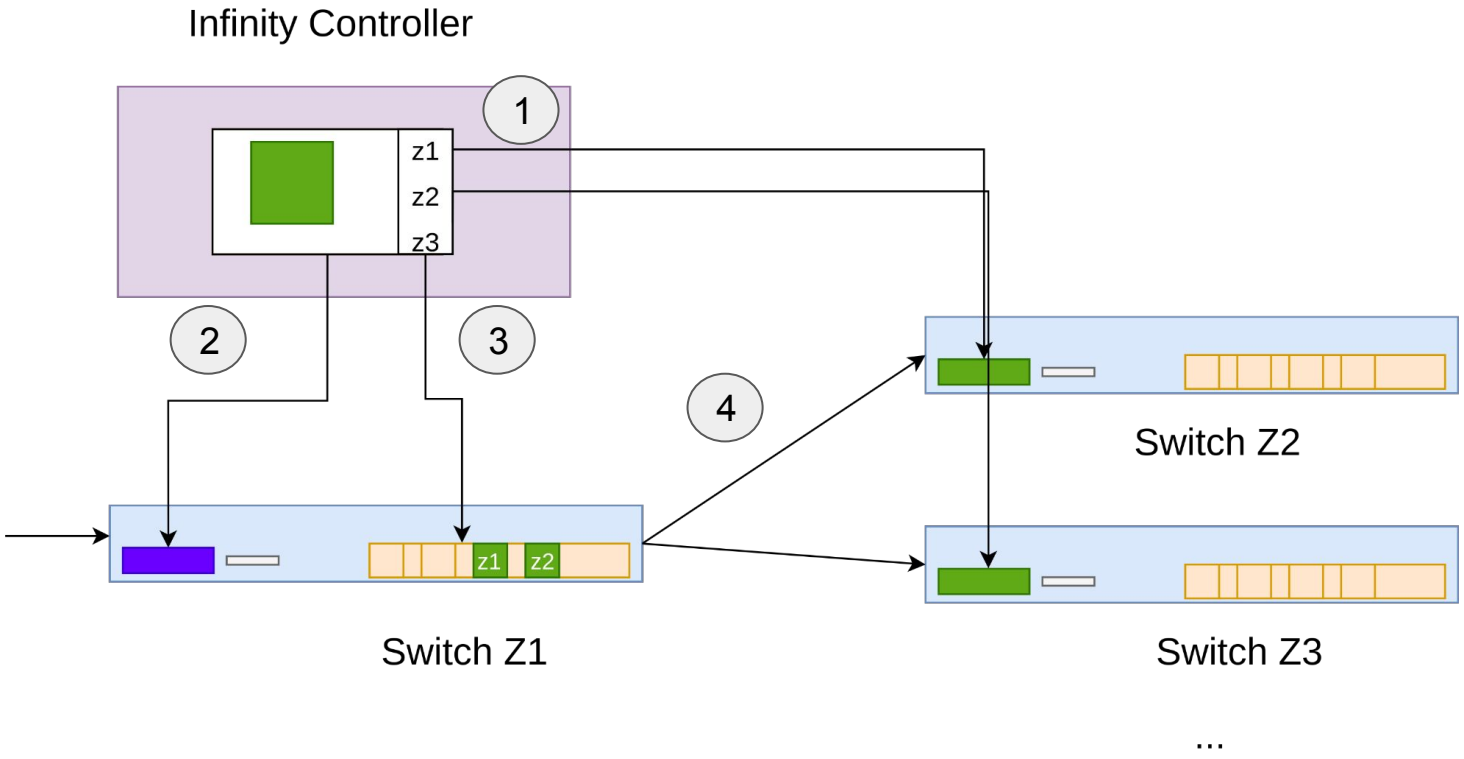
Distribute segments sequentially or horizontally

Expand processing stages, memory and bw

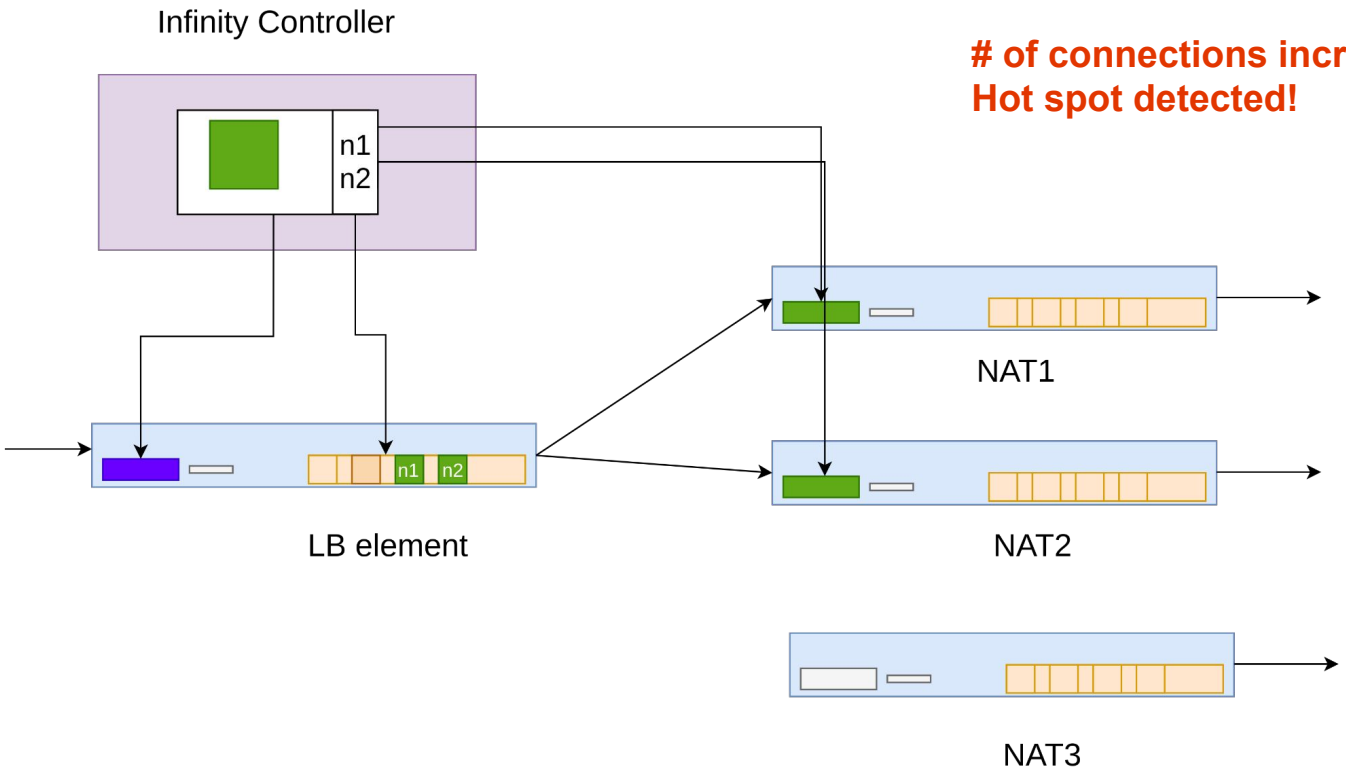
Primitive 1 - Sequential Decomposition



Primitive 2 - Horizontal Scaling

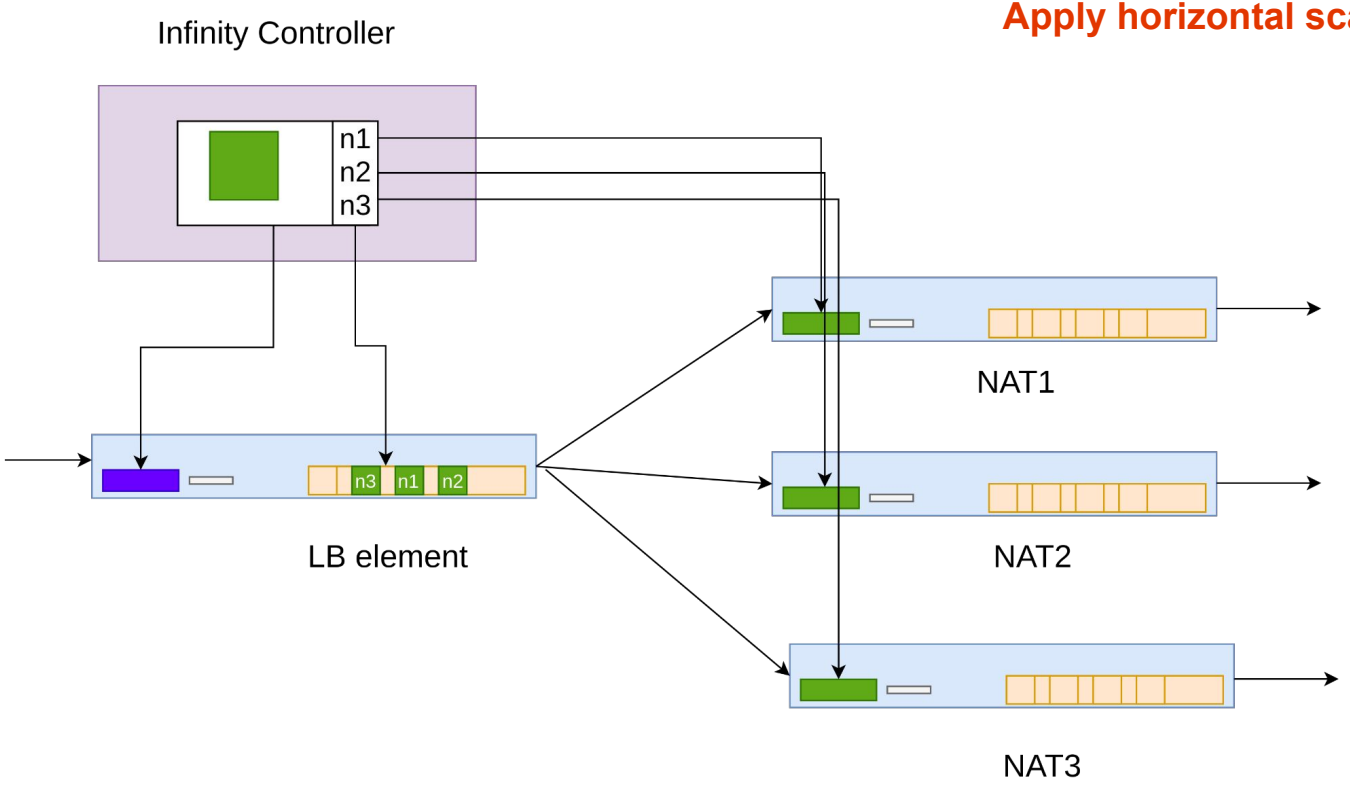


Example Use Case - Stateful NAT

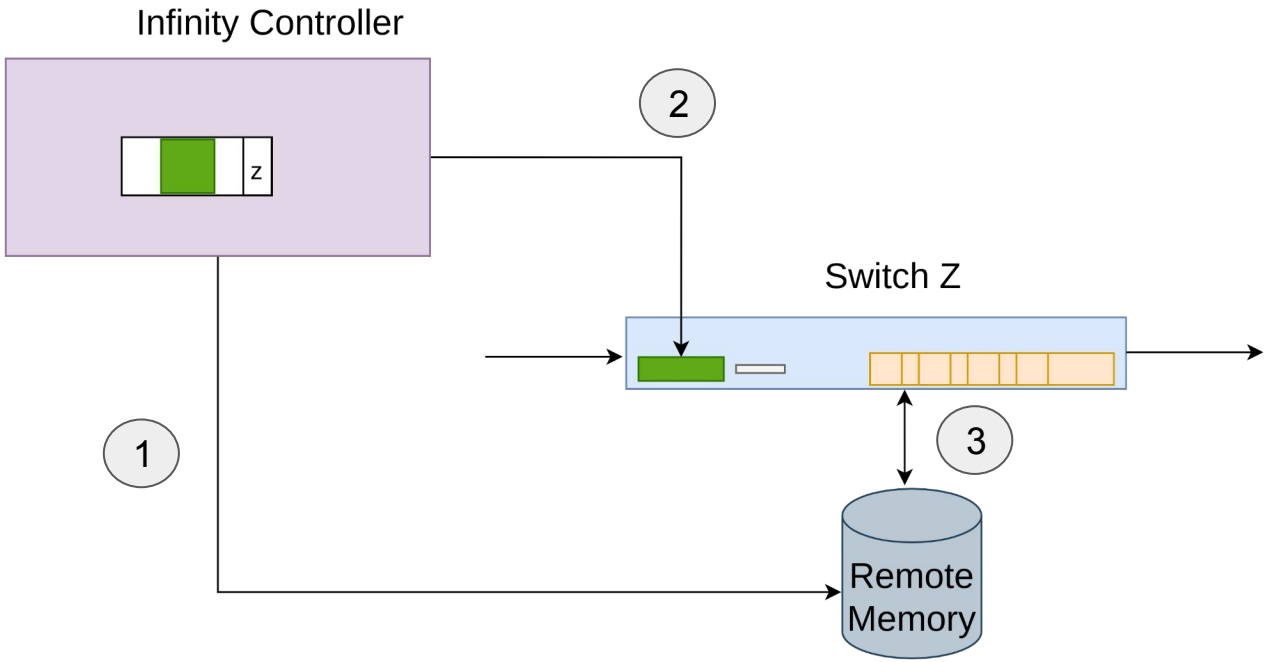


Example Use Case - Stateful NAT

Apply horizontal scaling primitive!



Primitive 3 - Vertical Scaling



Conclusion

- More flexible in-network app deployments
- Infinity primitives
- Challenges
- Work in progress
- Complete Infinity’s prototype
- Evaluate performance and scalability

Questions?