LIGHTWEIGHT 4OVER6

ONE STEP FURTHER DUAL-STACK LITE NETWORKS

Ripe 76, Marseille

dpino@igalia.com

igalia

AGENDA

- CGN & Dual-Stack Lite
- Lightweight 4over6
- Open-source implementations & adoption

IPV6 TRANSITION TECHNOLOGIES

Ease the transition to IPv6

Connectivity	Type of network	Transition technology
IPv4, IPv6	Dual-Stack	None
IPv6	IPv4-only	Tunnel broker/6in4, 6over4, 6rd, 6to4/Teredo, ISATAP, IVI/NAT64/SIIT
IPv4	IPv6-only	Dual-Stack Lite, Light weight 4over6, MAP-E, MAP-T, 4in6, 464XLAT, IVI/NAT64/SIIT

- IPv6-only: IPv4 as a service (*IPv4aaS*)
- Example: DT, Verizon, Comcast

CGN & DUAL-STACK LITE

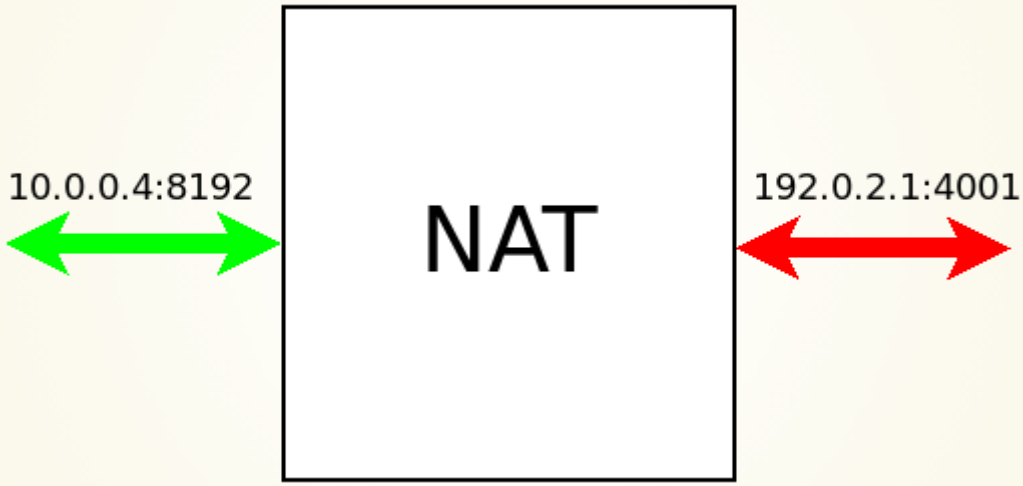
CARRIER-GRADE NAT

- IPv4 address exhaustion caught many off-guard
- NAT in the Carrier's network (a.k.a LSN or NAT444)
- Extends the pool of available IPv4 public addresses

NAT NETWORK FUNCTION

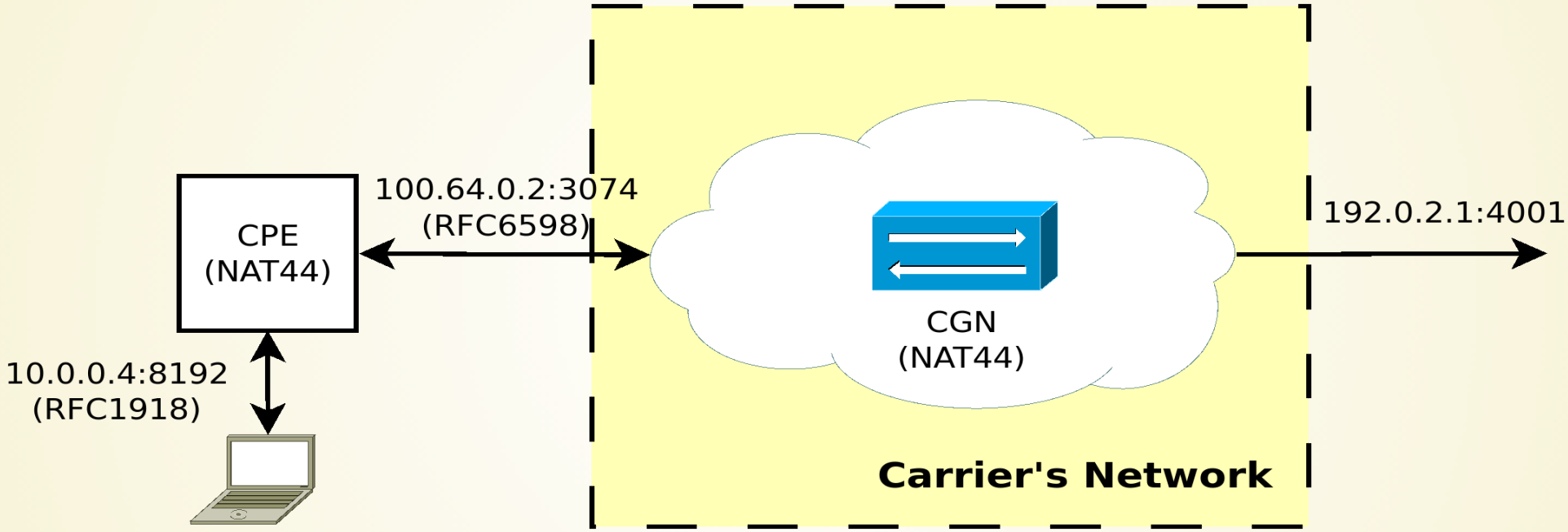
- Maps one address domain to another domain
- Stateful: Keeps per-flow information of the mapping
- Man-in-the-middle:
  - Breaks end-to-end principle of the Internet
  - Recalculate TCP & IP checksums

NAT NETWORK FUNCTION



{10.0.0.4, 8192} <=> {192.0.2.1, 4001}

CARRIER-GRADE NAT



{10.0.0.4, 8192}	<=>	{100.64.0.2, 3074}
{100.64.0.2, 3074}	<=>	{192.0.2.1, 4001}

INCONVENIENTS OF CGN

- NAT444:
  - Cannot host services
  - Breaks many applications
- Stateful: Hard to scale
- Many users share a public IPv4 address

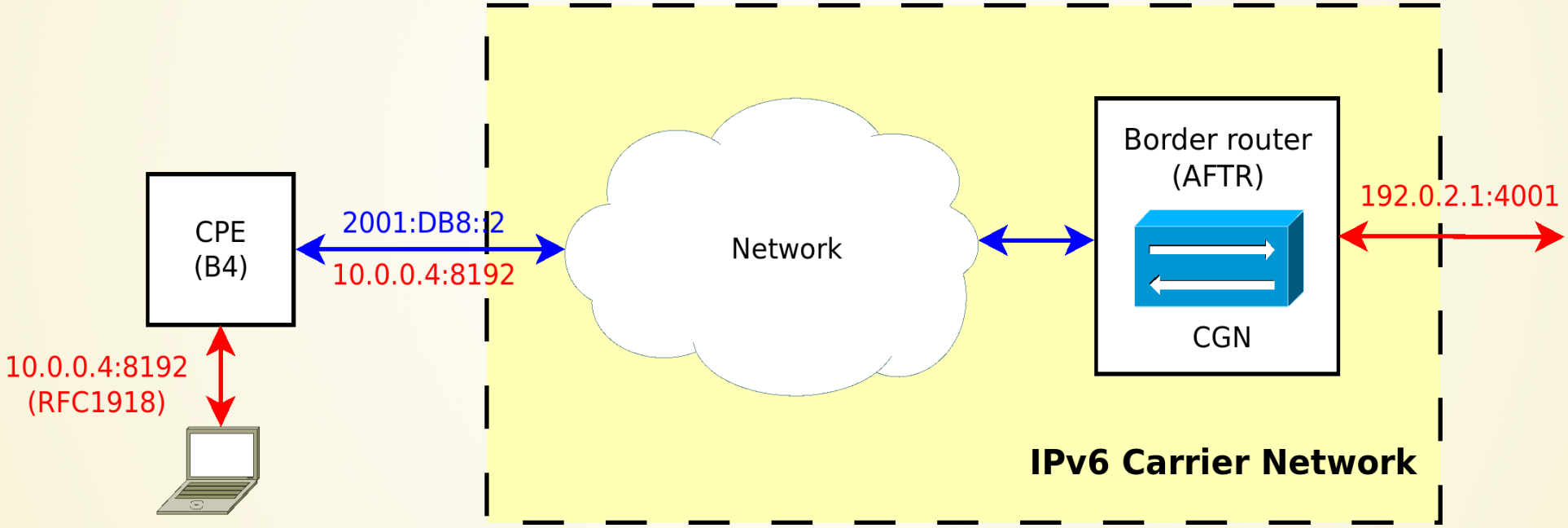
DUAL-STACK LITE

RFC 6333

DUAL-STACK LITE

- Carrier's network is IP6-only
- Provides IPv4 connectivity over IPv6
- A DS-Lite architecture consists of 2 end-points:
  - **B4**: Encap at CPE
  - **AFTR**: Decapsulation at Carrier + NAT

DUAL-STACK LITE



B4

- B4: *Basic Bridging BroadBand*
- Runs at CPE
- Tunnel end-point: encapsulates IPv4 in IPv6
- Acts as a DNS proxy

AFTR

- AFTR: *Address Family Translation Router*
- Runs at the Carrier's network
- Tunnel end-point: decapsulates IPv4-in-IPv6 packets
- Performs a NAT on the packet

IPv4 private	IPv4 public	IPv6 B4
{10.0.0.4, 8192}	{192.0.2.1, 4001}	2001:DB8::2

Binding-table

PROS & CONS

- Tunnel-based: tunnels can be anywhere
- Removes one level of NAT (CPE)
- IPv6-only network
- Handle fragmentation & reassembly
- Stateful

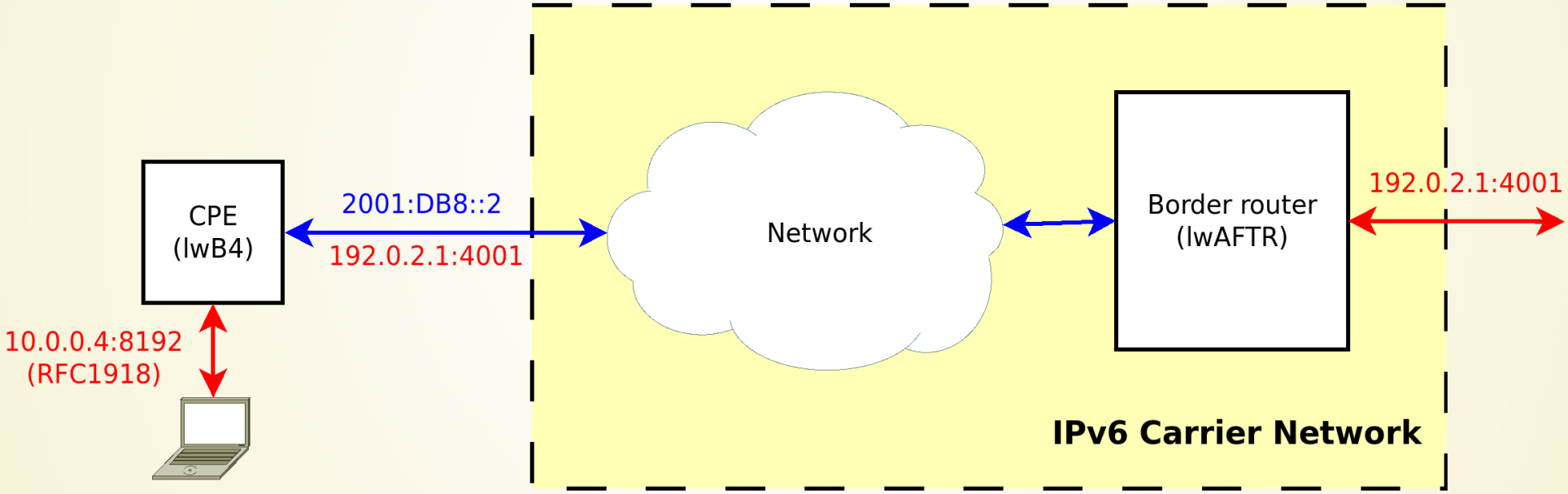
LIGHTWEIGHT 4OVER6

RFC 7596

LIGHTWEIGHT 4OVER6

- Moves back NAT to CPE
- It consists of 2 network functions:
  - **lwB4**: NAT + Encapsulation
  - **lwAFTR**: Softwire lookup + Decapsulation
- Relies on a A+P (Address + Port) softwires
- **A+P**: Split the port space into ranges of fixed size

LIGHTWEIGHT 4OVER6



SOFTWIRES

- Mappings between two different IP address families
- Technologies: MAP-E, MAP-T, lw4o6, 6rd...

IPv4 address	IPv4 port-range	IPv6 B4
192.0.2.1	8192-9215	2001:DB8::2

Binding-table

SOFTWIRES

datatracker.ietf.org/doc/draft-ietf-softwire-yang/

```
softwire-config {
  binding-table {
    softwire {
      ipv4 192.0.2.1;          // IPv4 public address.
      psid 8;                  // Port-set ID (8192-9215).
      b4-ipv6 2001:DB8::2;     // CPE's IPv6 address.
      br-address 2001:DB8::1;  // lwAFTR IPv6 address.
      port-set {
        psid-length 6;         // Number of port sets.
        psid-size 10;          // Size of a port set.
      }
    }
  }
}
```

SOFTWIRES

```
softwire {
  psid 8;                    // Port-set ID (8192-9215).
  port-set {
    psid-length 6;           // Number of port sets.
    psid-size 10;            // Size of a port set.
  }
}
```

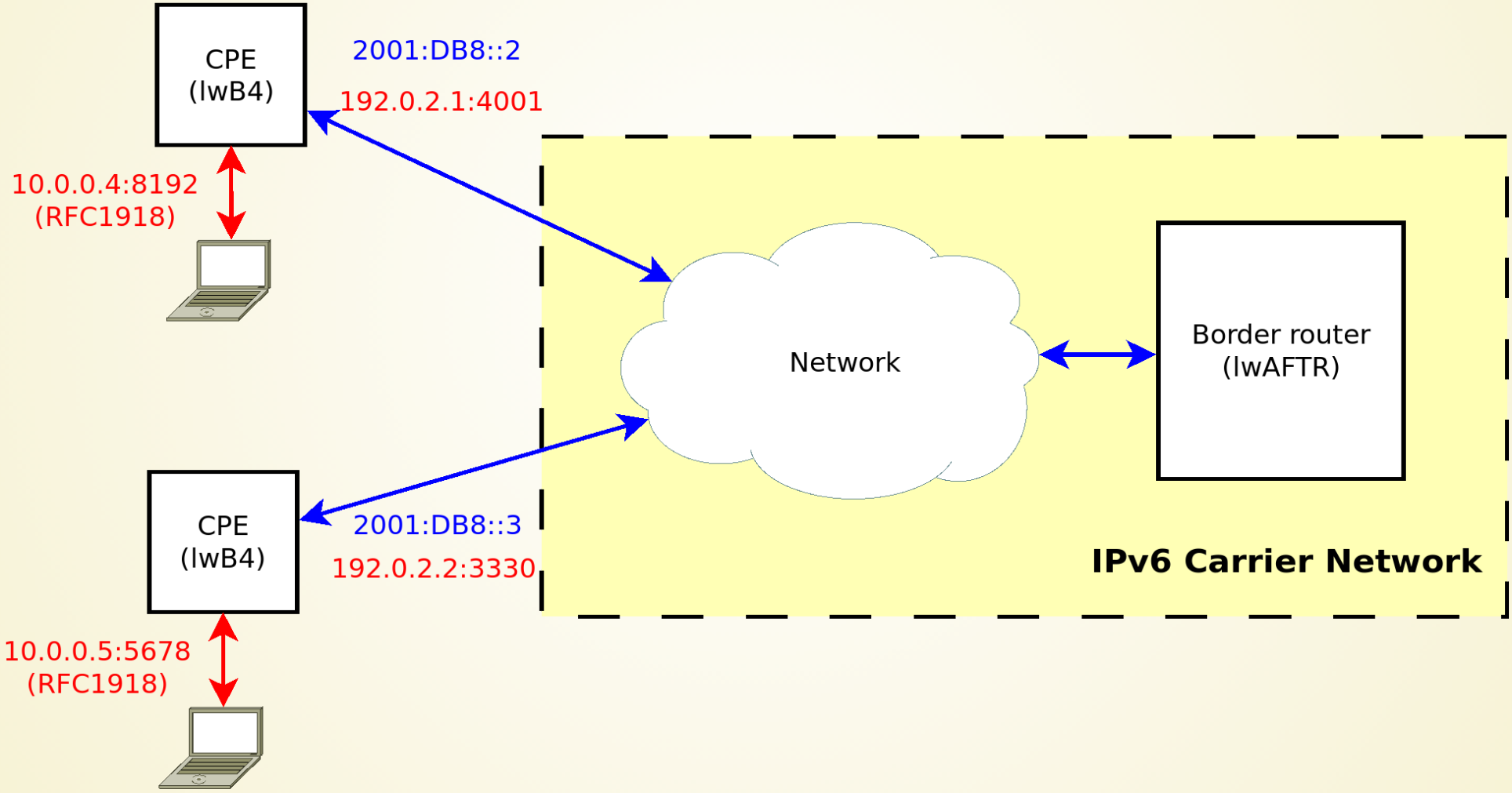
Port:	0-65535 (2^{16})
Port-set:	Length: 64 (2^{6}); Size: 1024 (2^{10})
Port-sets:	0-1023, 1023-2048,...,64511-65535
PSID:	8 ([8192-9215])

SOFTWIRES

```
softwire { ipv4 192.0.2.1; psid 8; port-set { psid-length 6; }
           br-address 2001:DB8::1; b4-ipv6 2001:DB8::2; }
softwire { ipv4 192.0.2.2; psid 0; port-set { psid-length 0; }
           br-address 2001:DB8::1; b4-ipv6 2001:DB8::3; }
softwire { ipv4 192.0.2.3; psid 1; port-set { psid-length 2; }
           br-address 2001:DB8::1; b4-ipv6 2001:DB8::4; }
```

Address	Port range
192.0.2.1	[8192-9215]
192.0.2.2	[0-65535]
192.0.2.3	[16384-32767]

HAIRPINNING



EXPERIMENTAL RESULTS

Source: Deployment considerations for lw4o6

App Type	Apps	Description	Result
Web	IE, Chrome	Browsing, download files	OK
Video	Youku, pptv, qqlive	VOD, live video	OK
P2P	Bittorrent, xunlei	Download files	OK
Ping/tracert	Command line	Ping/tracert URL	OK
TELNET/SSH	Putty, secureCRT	Telnet/SSH logging	OK
Email	126, QQ, hotmail	Send/receive email	OK
Cloud storage	Baidu Cloud	Upload/download files	OK
Instant messaging	Skype, QQ	Send/receive messages	OK
Online gaming	QQ game	Enter game	OK
Online payment	JD, Taobao	Complete payment	OK

PROS & CONS

- Removes the CGN
- Completely staleless
- Horizontal scaling
- Handle fragmentation & reassembly

LIGHTWEIGHT 4OVER6

OPEN-SOURCE IMPLEMENTATIONS & ADOPTION

lwB4

- Available in OpenWRT through MAP
- Emulated using Linux commands (*iproute2*, *iptables)*
- Marcel Wiget: Lw4o6 B4 Client in Linux namespace

lwAFTR

2 open-source implementations:

- Featured in VPP since v16.09
- Featured in Snabb since v15.10

Ole Trøan: Fast dataplanes with VPP (Fosdem 2018)

SNABB'S LWAFTR

SNABB

A toolkit for user-space networking development

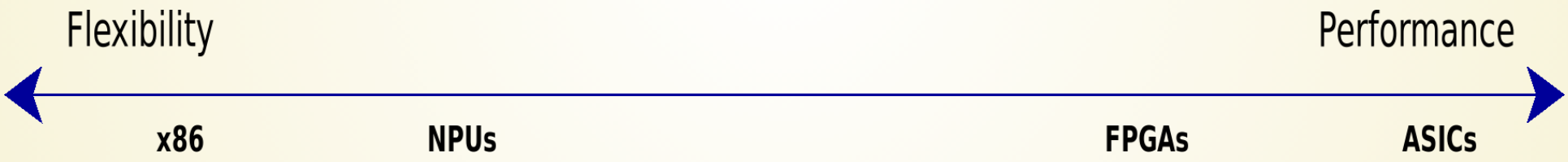
SNABB

- Started by Luke Gorrie (2012)
- Similar to DPDK (Intel) or VPP (Cisco)
- User space: kernel by-pass
- Written in Lua (LuaJIT)

https://github.com/snabbco/snabb

SOFTWARE DEFINED NETWORKING

Goal: programmable networking appliances



X86 NETWORKING

- NICs: 10G, 40G, 100G (Intel, Mellanox,...)
- PCIe v3 bandwith: 32 GB/s (@16 lanes)
- Multicore: 12-48 CPUs (Xeon)
- COTS network appliance

USER-SPACE NETWORKING

Linux networking layer cannot handle these speeds

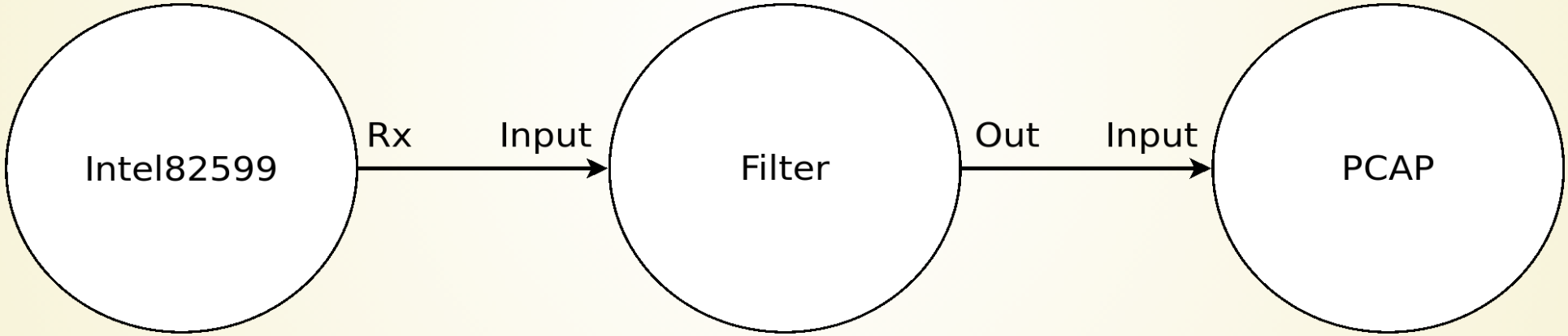
Update: XDP (eXtreme Data Path)

SNABB IN A NUTSHELL

Snabb consists of the following elements:

- **Apps**: represent a network function (Filter, lwAFTR)
- **Links**: connect apps together
- **Programs**: network graph of apps

SNABB IN A NUTSHELL



SNABB IN A NUTSHELL

- Snabb engine process packets in **breadths**
- A breadth consists of 2 steps:
  - ***Inhaling***, pulls packet from one or several sources and puts them in the links
  - ***Processing***, gives a chance to apps to do something with packets in their incoming links

SNABB IN A NUTSHELL

```
function run()
    local c = config.new()

    config.add(c, "nic", Intel82599, {pci="0000:04:00.0"})
    config.add(c, "filter", PcapFilter, "src port 80")
    config.add(c, "pcap", Pcap.PcapWriter, "output.pcap")

    config.link(c, "nic.tx -> filter.input")
    config.link(c, "filter.output -> pcap.input")

    engine.configure(c)
    engine.main()
end
```

SNABB IN A NUTSHELL

```
function PcapFilter:push ()
  local input = self.input.input
  local output = self.output.output

  while not link.empty(input) do
    local p = link.receive(input)
    if self:accept(p) then
      link.transmit(output, p)
    else
      packet.free(p)
    end
  end
end
```

VERSION 1.0

OCTOBER 2015

VERSION 1.0

- Prototype
- Basic functionality (encap/decap)
- Small binding-table (custom format)
- Performance: 2MPPS; 550-byte packet
- Developed own tools to measure performance
- End-to-end tests
- Other: fragmentation/reassembly, rate-limiting, ingress/egress filters

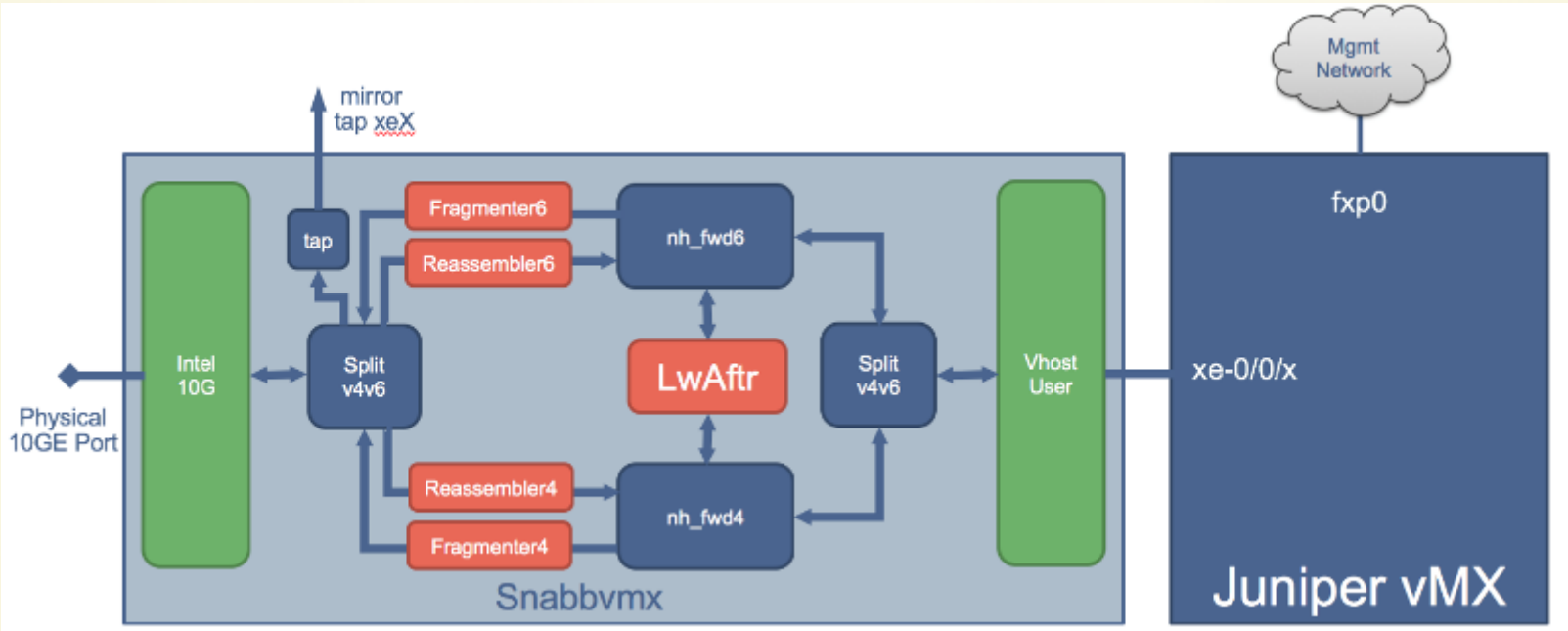
VERSION 2.0

FEBRUARY 2016

VERSION 2.0

- Full-compliant with RFC7596
- Large binding-table (custom format) (1M softwires)
- Performance: 2MPPS; 550-byte packet
- Control-plane: ping, arp, ndp
- Run as VNF
- Other: more end-to-end, CI, improved tools, sink-tests, VLAN tagging

SNABBVMX


Mgmt
Network
mirror
tap xeX
fxp0
tap
Fragmenter6
Reassembler6
nh_fwd6
Intel
10G
Split
v4v6
LwAftr
Split
v4v6
Vhost
User
xe-0/0/x
Physical
10GE Port
Reassembler4
Fragmenter4
nh_fwd4
Snabbvmx
Juniper vMX

HIGH-PERFORMANCE

- Profile heavily: no interpreted code on main loop
- Tight loops (busywait=true)
- Use SIMD (AVX): emit own assembly if neccessary
- Use HugePages (2MB) (Linux default is 4Kb)
- Disable Hyper-threading (no virtual CPUs)
- NUMA awareness
- Isolate CPUs (isolcpus=...)

VERSION 3.0

DECEMBER 2016

VERSION 3.0

- Added YANG support to Snabb
- Support of YANG IETF Softwire (draft-ietf-softwire-yang)
- New architecture: leader/worker
- Interoperability with external NETCONF-based control and management planes (Sysrepo)

VERSION 4.0

OCTOBER 2017

VERSION 4.0

- RSS (Received Side Scaling) support for Intel82599
- Multiprocess: 1 leader, n workers
- Support of YANG Alarm Module (draft-vallin-netmod-alarm-module)

ADOPTION

ADOPTION

- Terastream: DT next-generation network
- Testing: Croatia and Germany
- lw4o6 as IPv4aaS mechanism
- Production: Greece (OTE)
  - SnabbVMX: lwAFTR + Juniper VMX.
- Axel Claudberg: IPv6 in Terastream

CONCLUSIONS

- Lw4o6 is an iteration over DS-Lite
- Lw4o6 implements IPv4aaS in IPv6-only networks
- Relies on A+P softwires
- Stateless, easy to scale
- Full-fledged open-source implementations available

LINKS

- IPv6 deployment status and transition technologies
- A brief history of IPv4 address space exhaustion
- Dive into Lightweight 4over6
- IPv6 in Terastream
- Carrier Grade lw4o6 with Juniper vMX and Snabb
- RFC7596 - Lightweight 4over6
- RFC6333 - Dual-Stack Lite

THANK YOU!

Q&A

RIPE76, Marseille

https://github.com/igalia/snabb

dpino@igalia.com

igalia