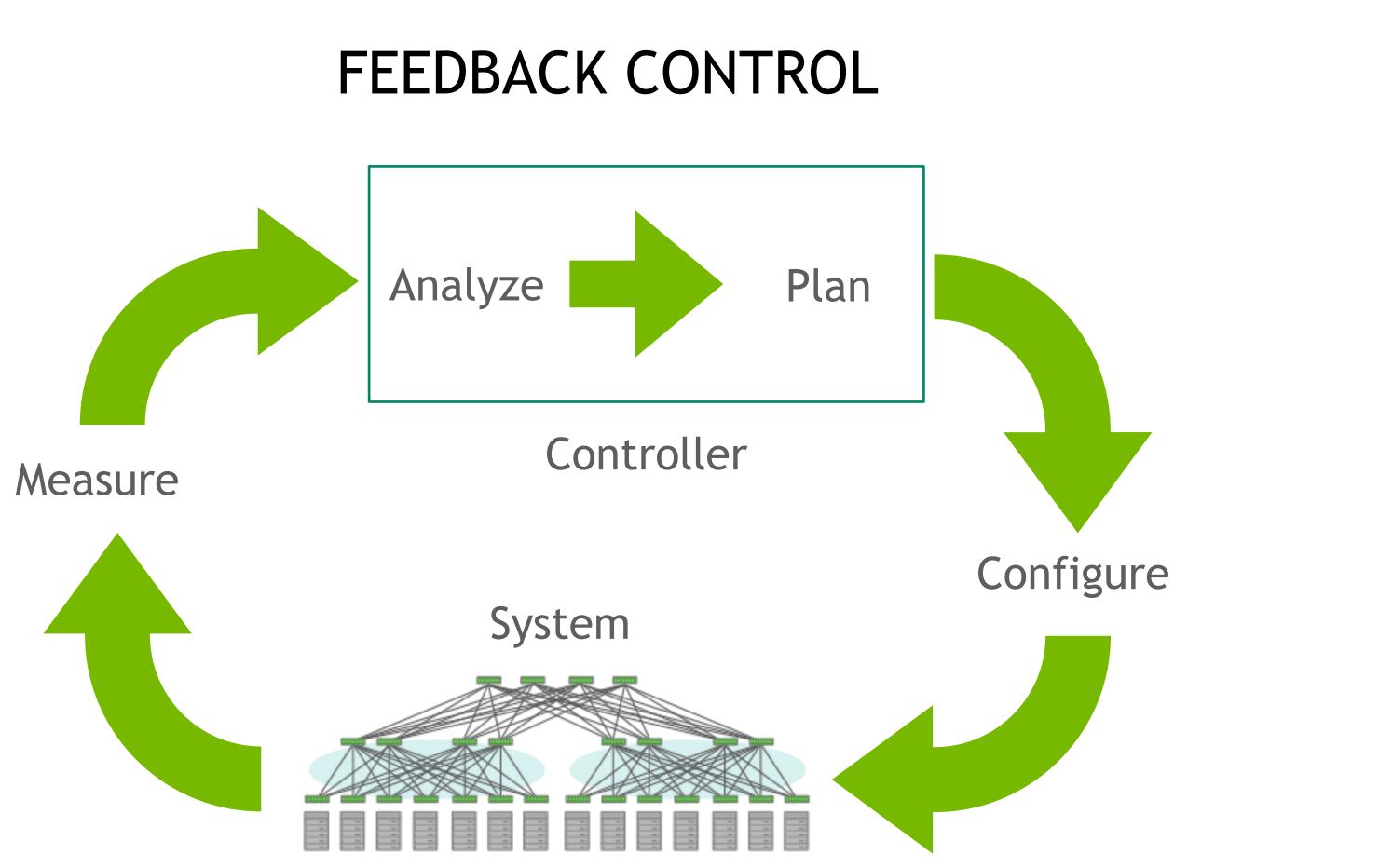
# "If you can't measure it, you can't improve it"



- Lord Kelvin

# WHY MEASURE?



## STABILITY REQUIREMENTS

### **Observable**

- Ability to monitor all important system states
- Example: To safely drive a car you need to see the road ahead, mirrors, and speedometer
- Controllable
  - Ability to influence all important system states
  - Example: To control a car's speed and direction you need to have a steering wheel, accelerator, and brakes
- Responsive
  - Feedback loop must be fast enough to track changes
  - Example: To avoid a car accident you need to react quickly if the car ahead brakes suddenly.

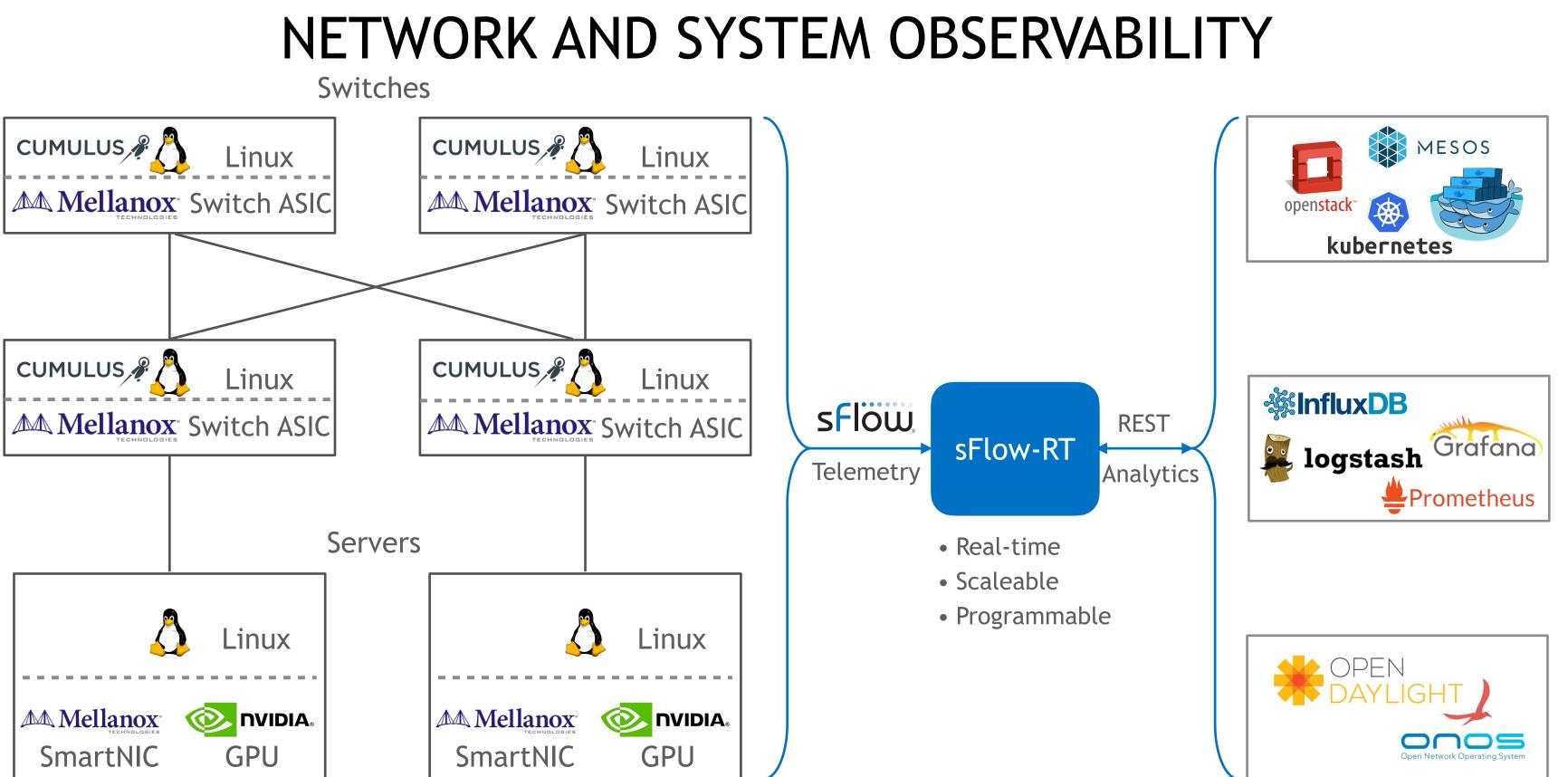


"In God we trust. All others bring data."



- Dr. Edwards Deming

# WHAT IS SFLOW?



Embedded monitoring of <u>all</u> switches, <u>all</u> servers, <u>all</u> applications, <u>all</u> the time

Consistent measurements shared between multiple management tools

## STANDARD COUNTERS

Network (maintained by hardware in network devices)

• MIB-2 ifTable: ifInOctets, ifInUcastPkts, ifInMulticastPkts, ifInBroadcastPkts, ifInDiscards, ifInErrors, ifUnkownProtos, ifOutOctets, ifOutUcastPkts, ifOutMulticastPkts, ifOutBroadcastPkts, ifOutDiscards, ifOutErrors

Host (maintained by operating system kernel)

- GPU: device count, processes, gpu time, mem time, mem total, mem free, ecc errors, energy, temperature, fan\_speed
- CPU: load\_one, load\_five, load\_fifteen, proc\_run, proc\_total, cpu\_num, cpu\_speed, uptime, cpu\_user, cpu\_nice, cpu\_system, cpu\_idle, cpu\_wio, cpu\_intr, cpu\_sintr, interupts, contexts
- Memory: mem\_total, mem\_free, mem\_shared, mem\_buffers, mem\_cached, swap\_total, swap\_free, page\_in, page\_out, swap\_in, swap\_out
- Disk IO: disk\_total, disk\_free, part\_max\_used, reads, bytes\_read, read\_time, writes, bytes\_written, write\_time
- Application (maintained by application)
  - HTTP: method\_option\_count, method\_get\_count, method\_head\_count, method\_post\_count, method\_put\_count, method\_delete\_count, method\_trace\_count, method\_connect\_count, method\_other\_count, status\_1xx\_count, status\_2xx\_count, status\_3xx\_count, status\_4xx\_count, status\_5xx\_count, status other count

# SFLOW'S SCALABLE PUSH PROTOCOL

### Simple

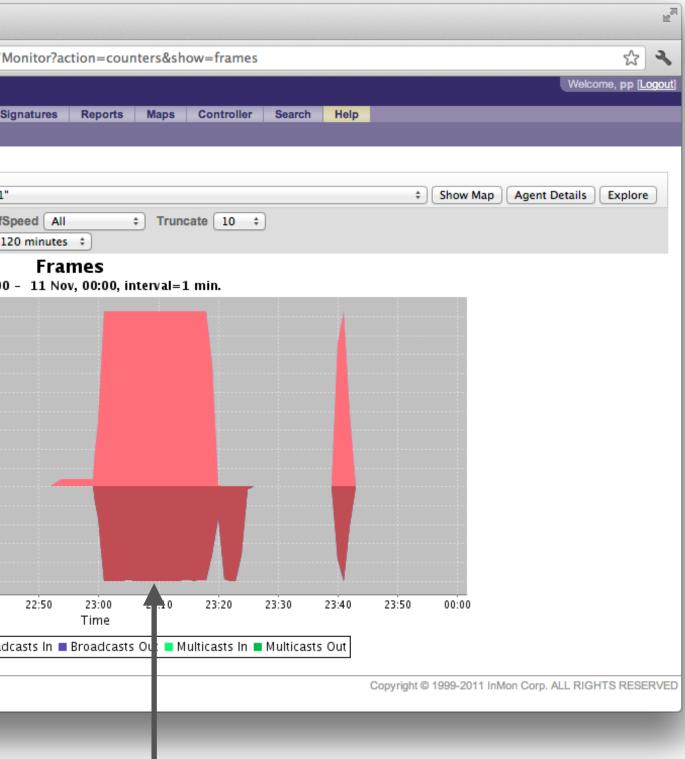
- standard structures containing densely packed blocks of counters
- extensible (tag, length, value)
- RFC 1832: XDR encoded (big endian, quad-aligned, binary) simple to encode/decode
- unicast UDP transport
- Minimal configuration
  - collector address
  - polling interval
- Cloud friendly
  - flat, two tier architecture: many embedded agents  $\rightarrow$  central "smart" collector
  - sFlow agents automatically start sending metrics on startup, automatically discovered
  - eliminates complexity of maintaining polling daemons (and associated configurations)



### **COUNTERS AREN'T ENOUGH** Don't tell you why there is a spike

- Counters tell you there is a problem, but not why.
- Counters summarize performance by dropping high cardinality attributes:
  - ip addresses
  - protocols
  - ports
- Need to be able to efficiently disaggregate counter by attributes in order to understand root cause of performance problems
- How do you get this data when there are millions of packets per second?

nMon	Traffi	c Se	ntinel		
	ome Event Eventerfaces T		Traffic Factors	Hosts Circles	Service Top N
	iteriaces   <u>i</u>	rend	raciois	GILCIES	
Filter:					
SC11 >	Seattle > Seattle	Cinet >	<u>Core</u> > <u>c</u>	<u>ore-1</u> > (	"edge-2:
Show	Frames	÷ if	Туре 🖪	11	
Date	Thu 11/10	÷T	'ime 🚺	:00 PM	+ Inter
					10 Nov,
					10 100,
9 N					
8 N 7 N					
6 N					
	4				
	1				
	1				
	1 - 1 -				
	1 - 1 - 1 -				
	1 - 1 - 1 -				
Frames per Second Frames per Second N Frames Per Second N Frames Per Second	11 - 11 - 11 - 11 - 11 - 11 -				
Frames per Second N 2 N 1 N 1 N 1 N 2 N 2 N 2 N	1 - 1 - 1 - 1 - 1 - 1 -				
5 N 4 N 3 N 2 N 4 C 2 N -2 N -3 N -3 N	1 - 1 - 1 - 1 - 1 - 1 - 1 -				
Frames per Second N 2 N 1 N 1 N 1 N 2 N 2 N 2 N	11 - 11 - 11 - 11 - 11 - 11 - 11 - 11 -				
Frames per Second N 2 Per Second N 1 N - 1 N - 3 N - 4 N - 4 N	11 - 11 - 11 - 11 - 11 - 11 - 11 - 11 -	22:10	22:2	) 22	:30 2
5 N 8 N 8 N 1 9 N 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1	22:10	22:21	) 22	:30 2

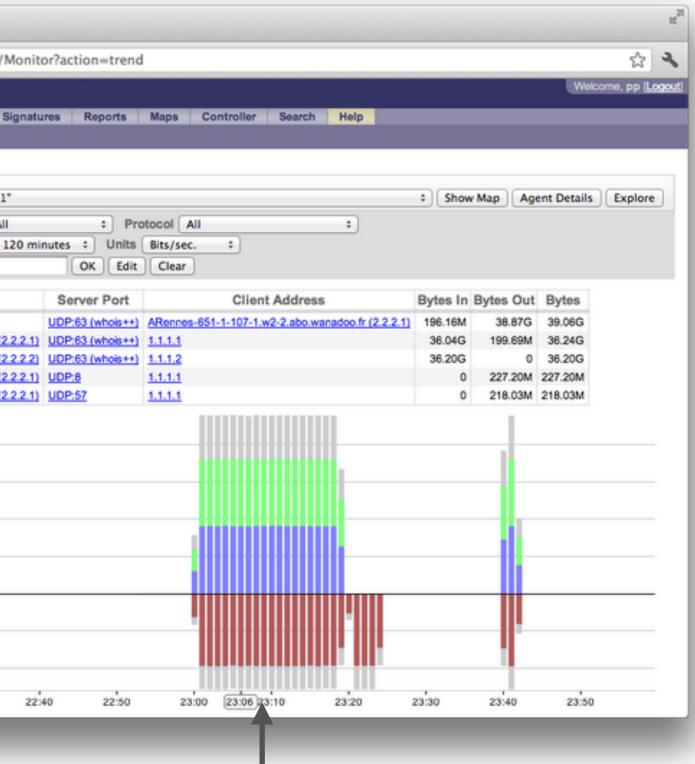


- Why the spike in traffic?
- (100Gbit link carrying 14,000,000 packets/second)

### **RANDOM SAMPLING** Scaleable method of adding details

- Random sampling is lightweight
- Unbiased results with known accuracy
- Sampled packet header (128 bytes)
- Forwarding state associated with sampled packet (e.g. ingress/egress port, VLAN, next hop, CIDR etc.)
- Linux kernel instrumenation randomly samples packets
- Offload to hardware instrumentation on network switch scales functionality to Terabit/s speeds
- Identify top sources, destinations, connections, protocols

	Traffic 5	entinel		×
← → 0	* ff 📧	https://in	mon.sc1	1.org/inmsf/
inMon	Traffic S	entine	əl 🛛	
File Hom	e Events	Traffic	Hosts	Services
Status   Inter	faces   Trend	Factors	Circles	Top N
Filter:				
	attle > SCine	t > Core >	<u>core-1</u> > (	"edge-2:2/1/
Chart T	op Connectio	ns		+ Host A
Date Th	nu 11/10 💠	Time 1	0:00 PM	Interval
Where (?	)			
		5	Server Ad	dress
	1.1.1	1		
			07-1.w2-2.a	bo.wanadoo.fr (
	ARen	nes-651-1-1	07-2.w2-2.a	ibo.wanadoo.fr (
	ARen	nes-651-1-1	07-1.w2-2.a	ibo.wanadoo.fr (
	ARen	nes-651-1-1	07-1.w2-2.a	ibo.wanadoo.fr (
80G				
60G				
ष्टु 40G				
puopeg and still				
ts b				
° •				
20G				
40G				
l	22	10	22:20	22:30



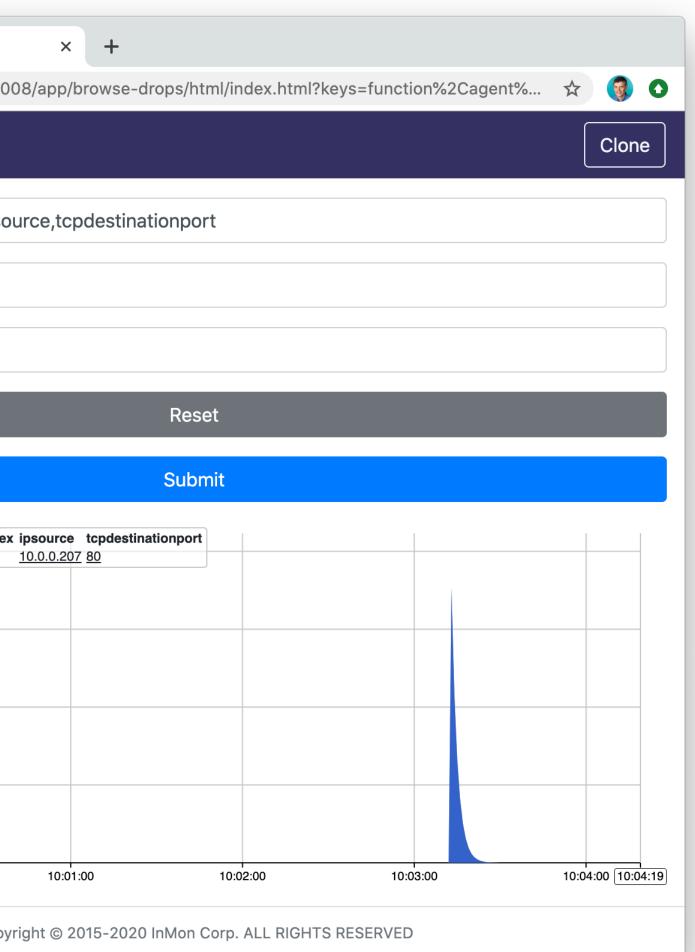
Break out traffic by client, server and port

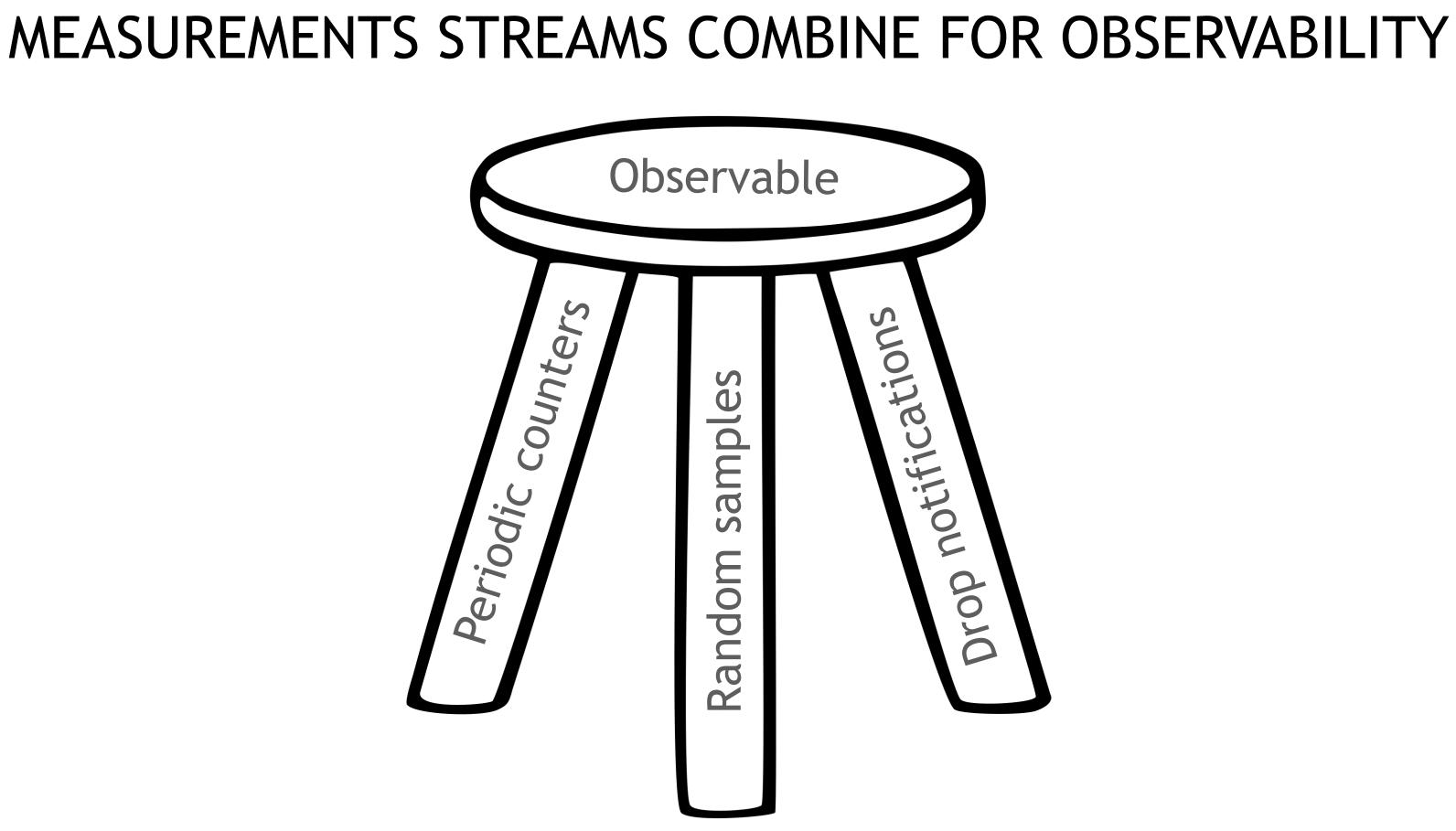
(graph based on samples from100Gbit link carrying 14,000,000 packets/second)

### **NEW! PACKET DROP EVENTS** Details of rare but critical drop events

- Discard counters tell you that packets are being dropped, but not root cause
- Packet discards are rare and unlikely to be sampled
- Packet discards can severely impact performance and availability of critical services
- Packet discard notifications in sFlow:
  - Discarded packet header (128 bytes)
  - Reason for dropping packet
- Linux kernel netlink drop\_monitor API reports each dropped packet and the reason it was dropped
- Offload to hardware instrumentation on a network switch scales functionality to Terabit/s network speeds
- Identify sources, destinations, protocols, locations of discards

	6	Disca	ard B	rowser	
$\leftarrow \rightarrow$	C		i	local	host:80
inM	on	Disc	ard	Bro	wser
func	tion,ag	gent,i	npu	tifind	ex,ipsc
fps					
Filte	r				
0.4	fun	ction	agen	nt in	putifinde
0.4	<u>tcp</u>	<u>v4_rcv</u>	<u>10.0</u> .	<u>0.50</u> <u>2</u>	
0.3 ح					
Frames per Second					
ames pe	2				
ی ۳ 0.1					
C	,		10:00	:00	
					Сору
					000)





sFlow's integrated data model combines measurement streams for comprehensive view

### SFLOW IS AN INDUSTRY STANDARD Broad multi-vendor support ensures observability of physical network



CISCO





IT agility. Your way.\*

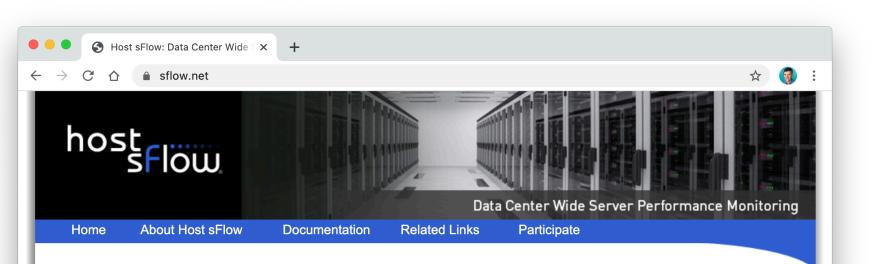
**Hewlett Packard** Enterprise











The Host sFlow agent exports physical and virtual server performance metrics using the sFlow protocol. The agent provides scalable, multi-vendor, multi-OS performance monitoring with minimal impact on the systems being monitored.

### NEWS

July 30, 2020 - Netlink dropmon support implemented

September 6, 2019 - SONiC support implemented

- April 1, 2019 Netlink psample support implemented
- September 14, 2018 Systemd traffic marking
- November 27, 2017 OpenSwitch support implemented
- March 29, 2017 Arista EOS support implemented
- December 14, 2016 Systemd service monitoring
- October 14, 2016 Linux tcp diag support implemented (delay, loss, jitter, bandwidth)
- August 20, 2016 Dell OS10 support implemented
- March, 30, 2016 SFP optical monitoring added
- February 26, 2016 BPF and PCAP monitoring added
- September 16, 2015 NFLOG monitoring added, supports Cumulus VX (virtual) switches
- February 24, 2015 MIB2 ip,icmp,tcp,udp + Broadcom ASIC table counters added
- December 4, 2014 CPU steal, guest and guest nice metrics added



Supported operating systems:

- AIX
- FreeBSD
- Linux
- Solaris
- Windows
- Supported Linux containers:
- Docker
- Systemd
- Supported hypervisors:
- Hyper-V
- KVM/libvirt
- Nutanix AHV
- Xen/XCP/XenServer
- Supported switches:
- Arista EOS
- Cumulus Linux
- Dell OS10
- OpenSwitch
- SONiC

### HOST SFLOW Extends observability into software network edge





















### Configure sFlow agents Edit /etc/hsflowd.conf

Host sFlow agent pre-installed on Cumulus Linux Download server package from <u>https://sflow.net</u> Same software and configuration on switches and servers minimises operational complexity # Minimal configuration
sflow {
 collector { ip = 10.31.234.46 }
}

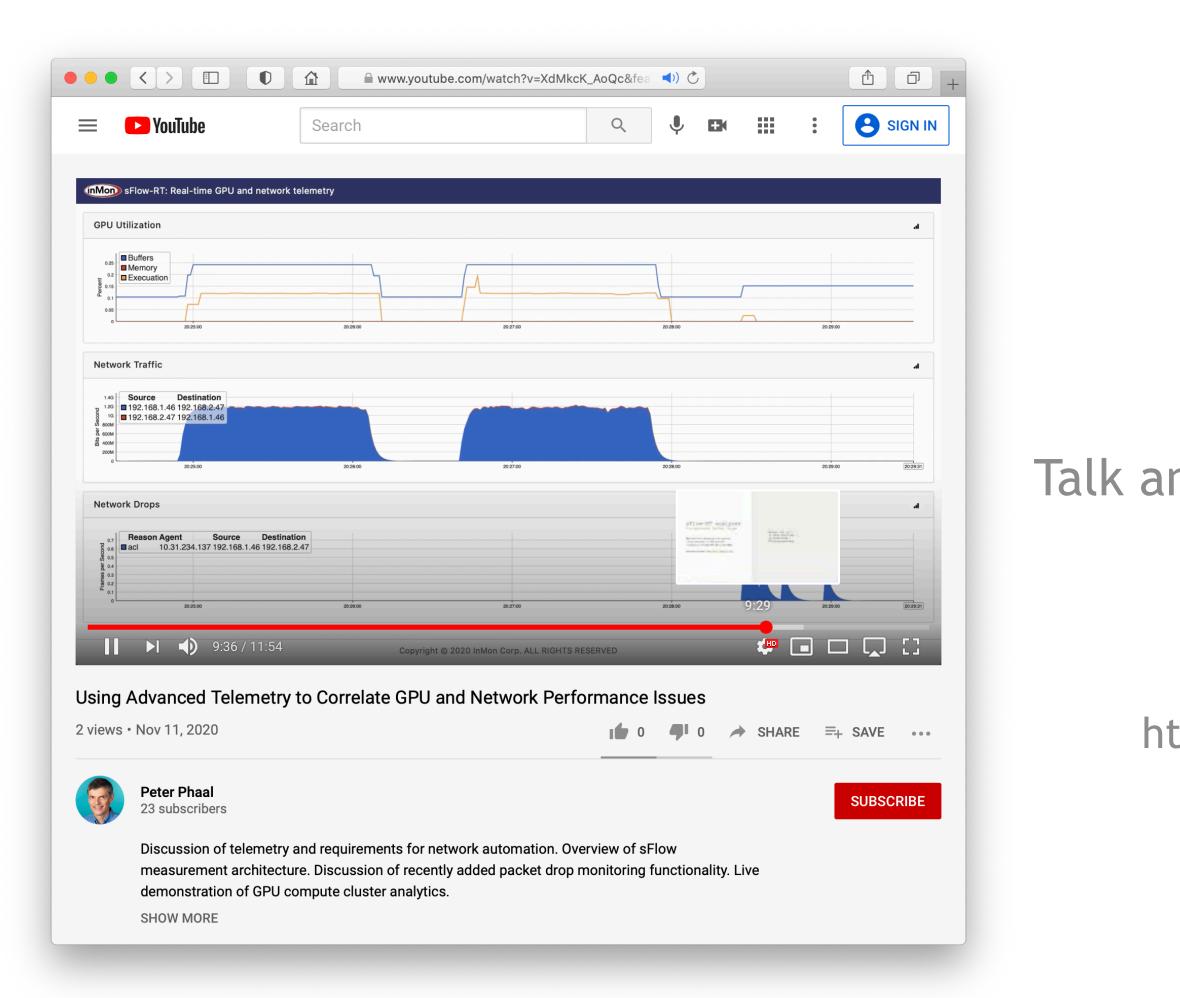
### sFlow-RT analyzer Pre-packaged Docker image

Real-time sFlow analytics as a microservice:

- sFlow telemetry in on UDP port 6343
- Analytics out through REST API on port 8008

Download packages from <a href="https://sflow-rt.com">https://sflow-rt.com</a>

docker run -d \
-p 6343:6343/udp \
-p 8008:8008 \
sflow/prometheus



### Talk and demo available on YouTube

### https://youtu.be/XdMkcK\_AoQc