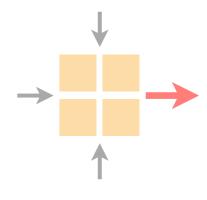
# Advanced Topics in Communication Networks Programming Network Data Planes



Laurent Vanbever nsg.ee.ethz.ch

ETH Zürich Nov 1 2018

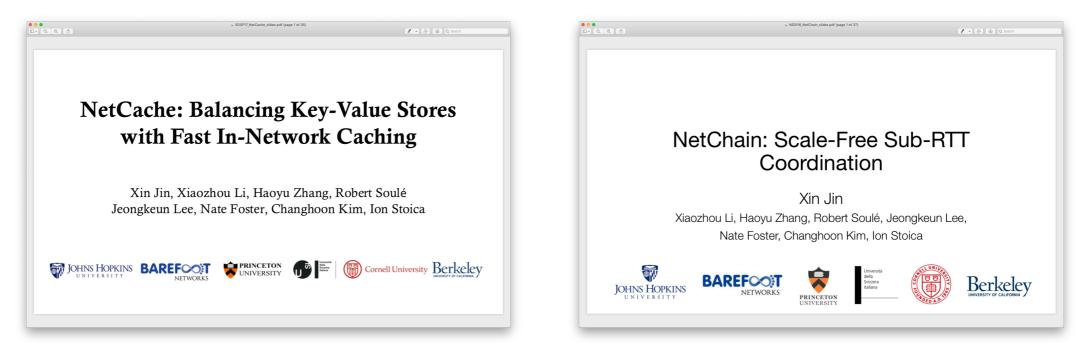


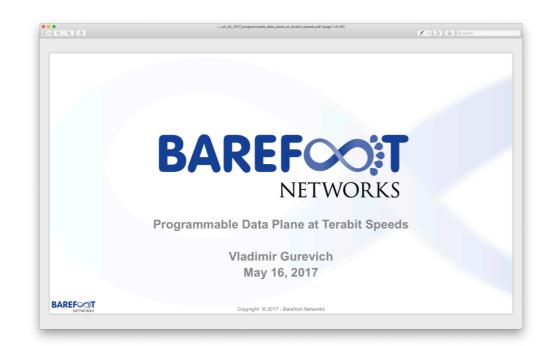
#### Last week on

### Advanced Topics in Communication Networks

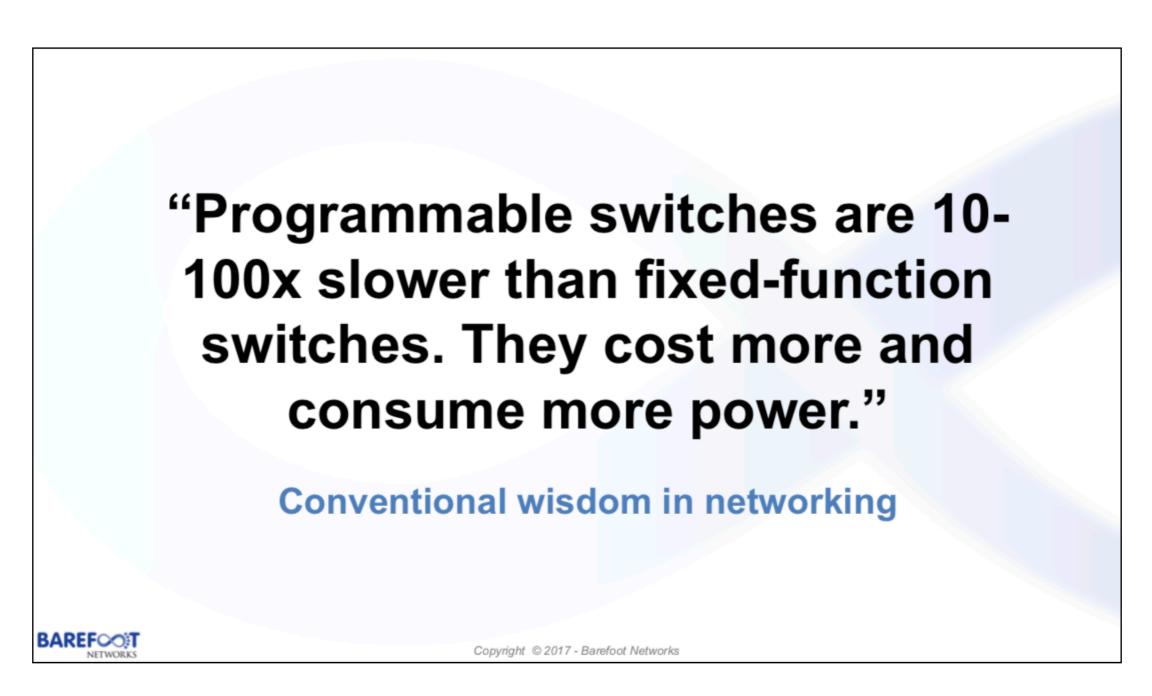
We looked at the Tofino architecture together with two (key, value) store applications: Net/{Cache, Chain}



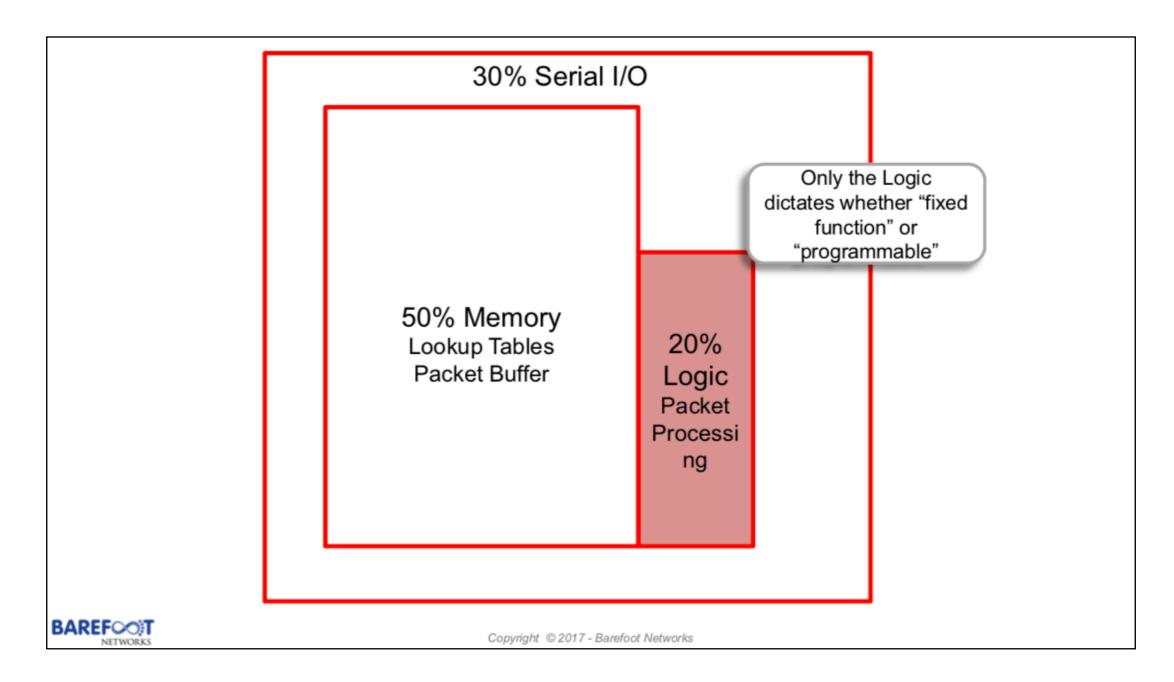






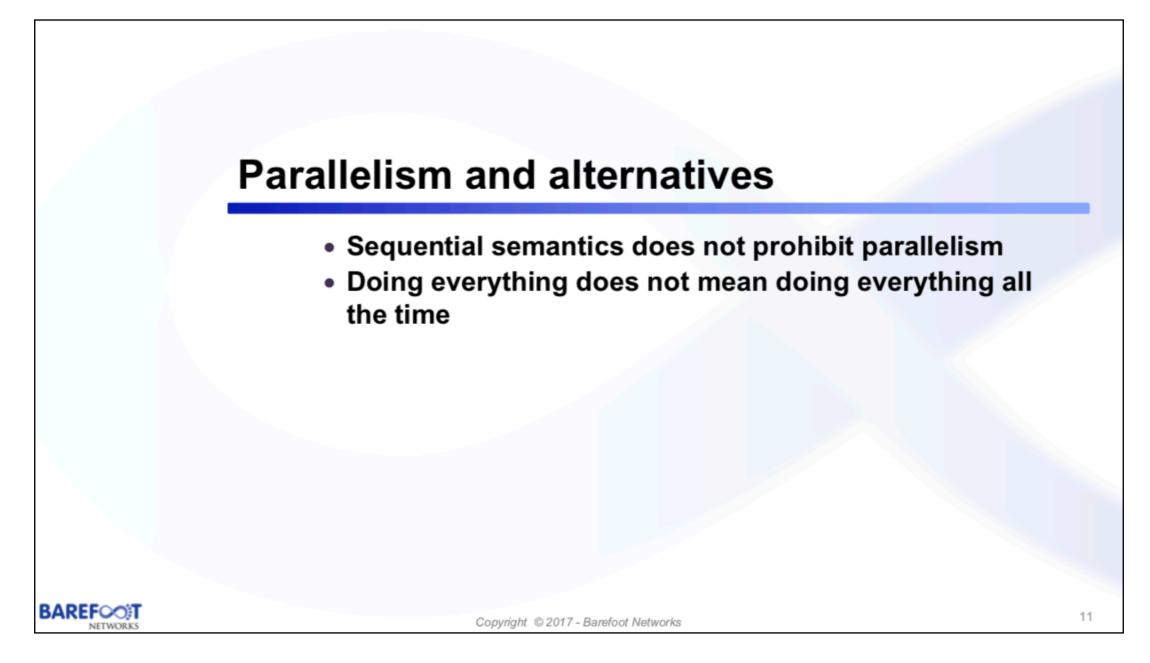


One of the main enabler for data-plane programmability is the shrinking size of the packet processing logic chip.



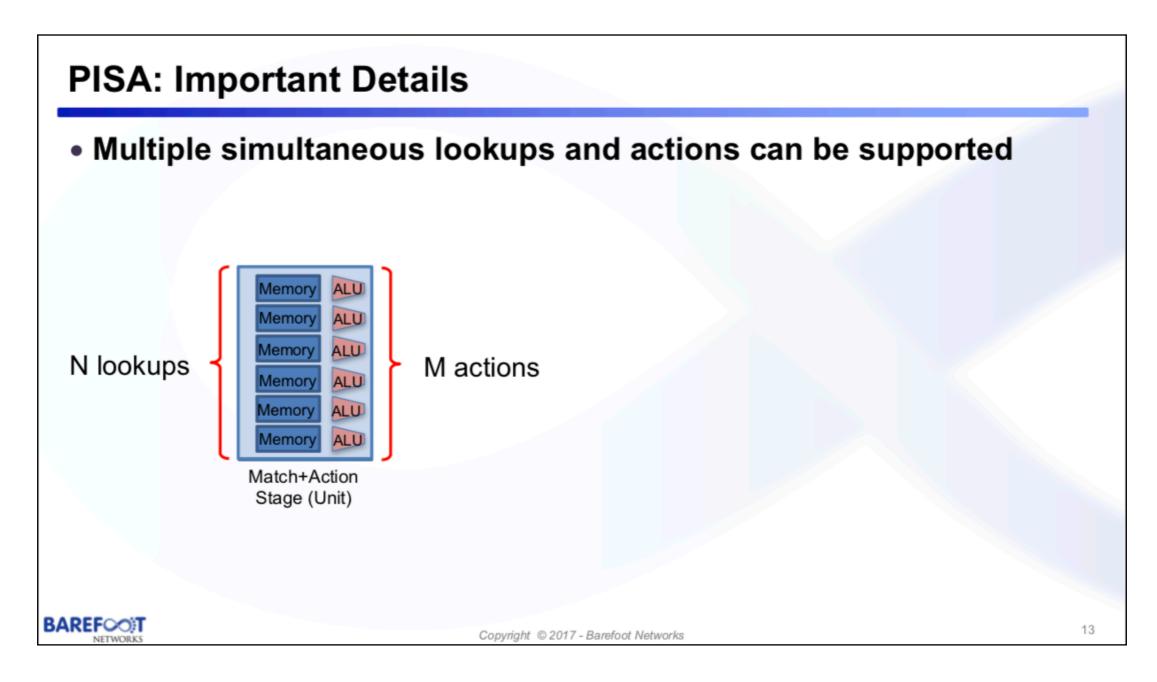
# Barefoot Tofino processes packets in parallel,

even though the semantic of a P4 program is sequential



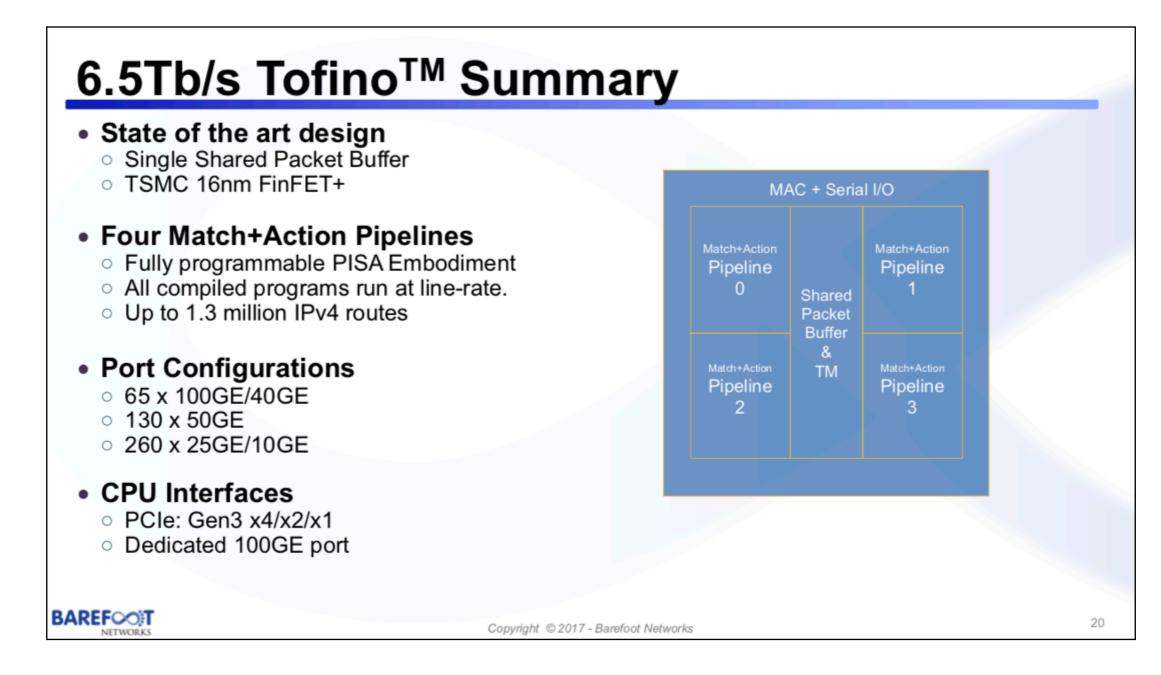
## Barefoot Tofino processes packets in parallel,

even though the semantic of a P4 program is sequential

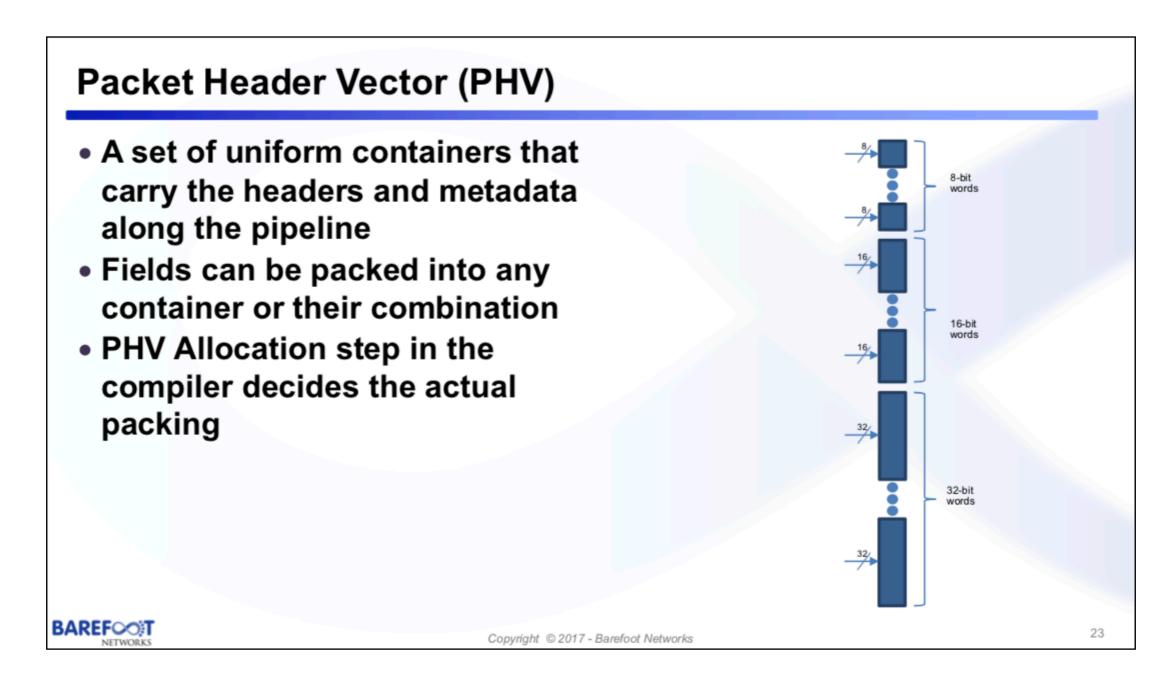


# Barefoot Tofino 6.5 Tbps backplane

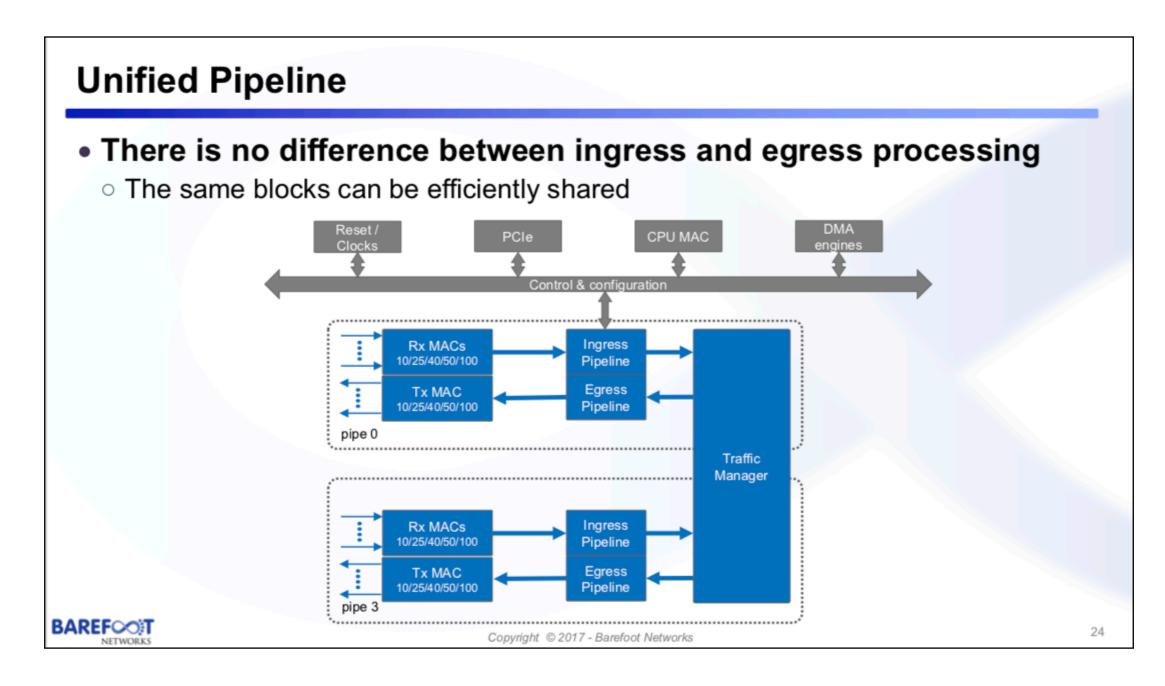
several billion packets per second at line rate



Tofino relies on Packet Header Vector (PHV) to pass states between stages—this is one of the limiting factor



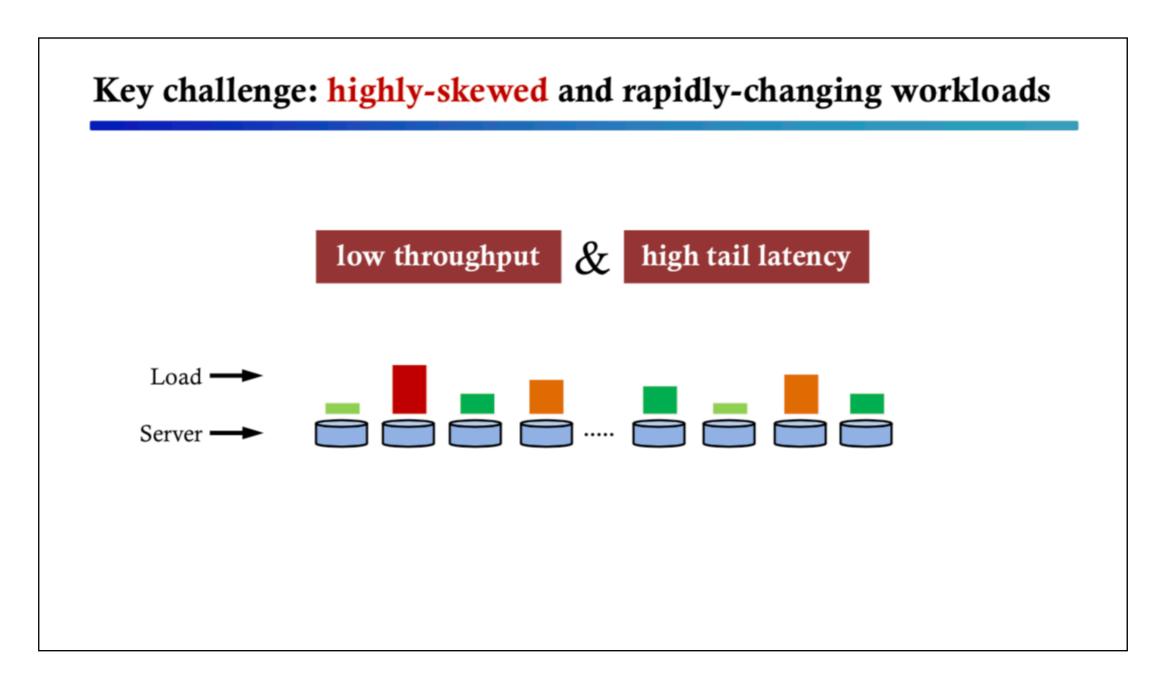
Tofino uses a folded pipeline in which the *same* stages are used for both the ingress and the egress pipeline





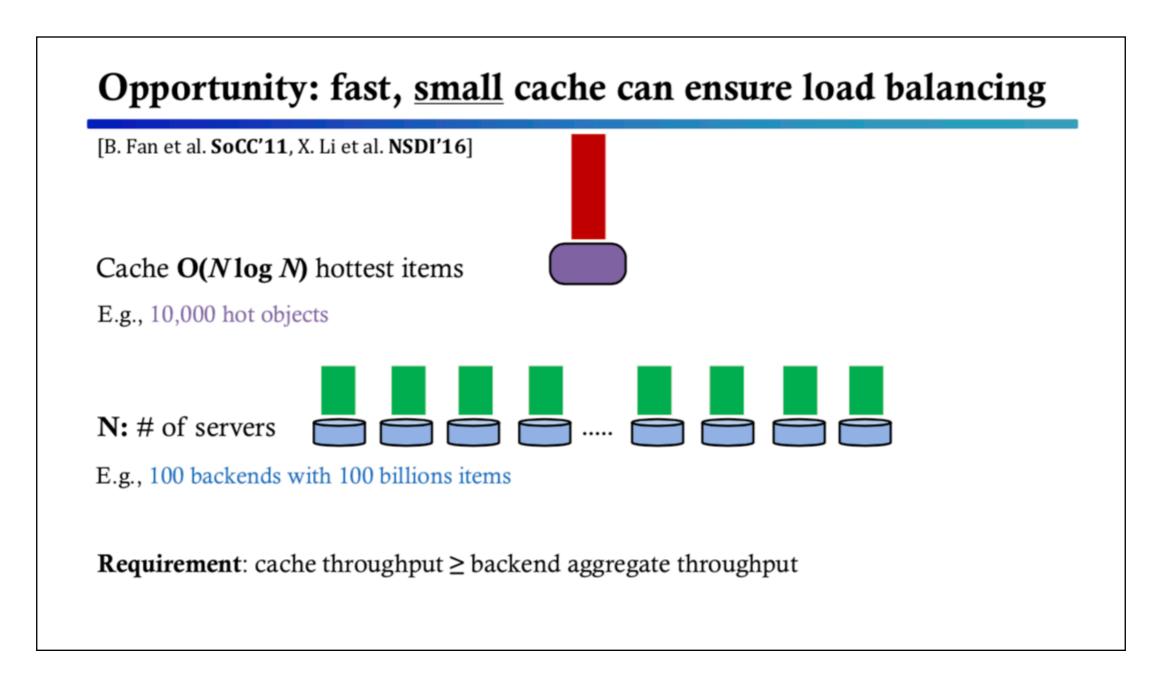


NetCache solves the problem of load-balancing in key-values stores observing *dynamic*, *skewed* workload



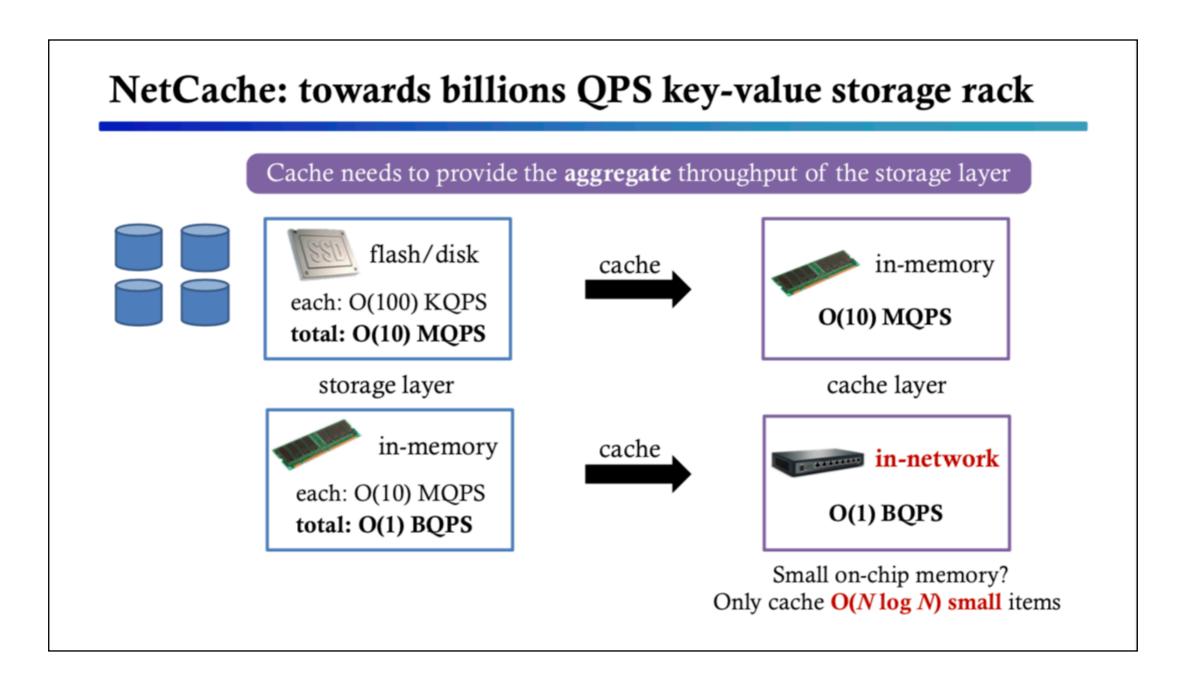
Source: NetCache: Balancing Key-Value Stores with Fast In-Network Caching, Xin Jin, 2017

It leverages that a small but very fast cache can provide perfect load-balancing... in theory



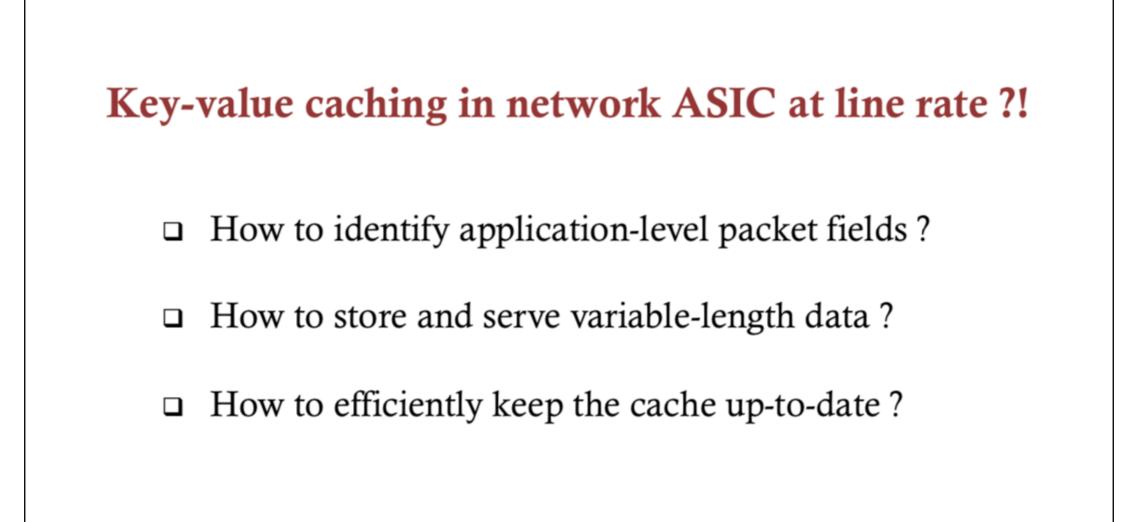
Source: NetCache: Balancing Key-Value Stores with Fast In-Network Caching, Xin Jin, 2017

NetCache relies on the O(billion) throughput of programmable network devices to achieve it in practice

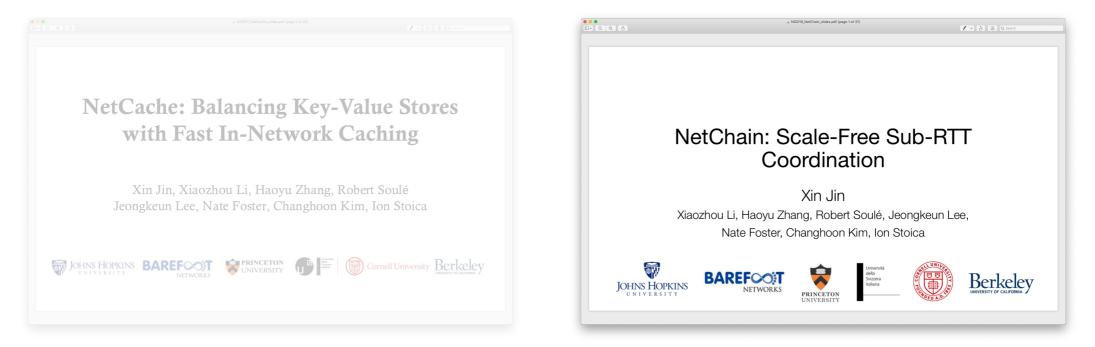


Source: NetCache: Balancing Key-Value Stores with Fast In-Network Caching, Xin Jin, 2017

It relies on a tailored UDP-based protocol, an de/encoding scheme for storing variable length values, and sketches







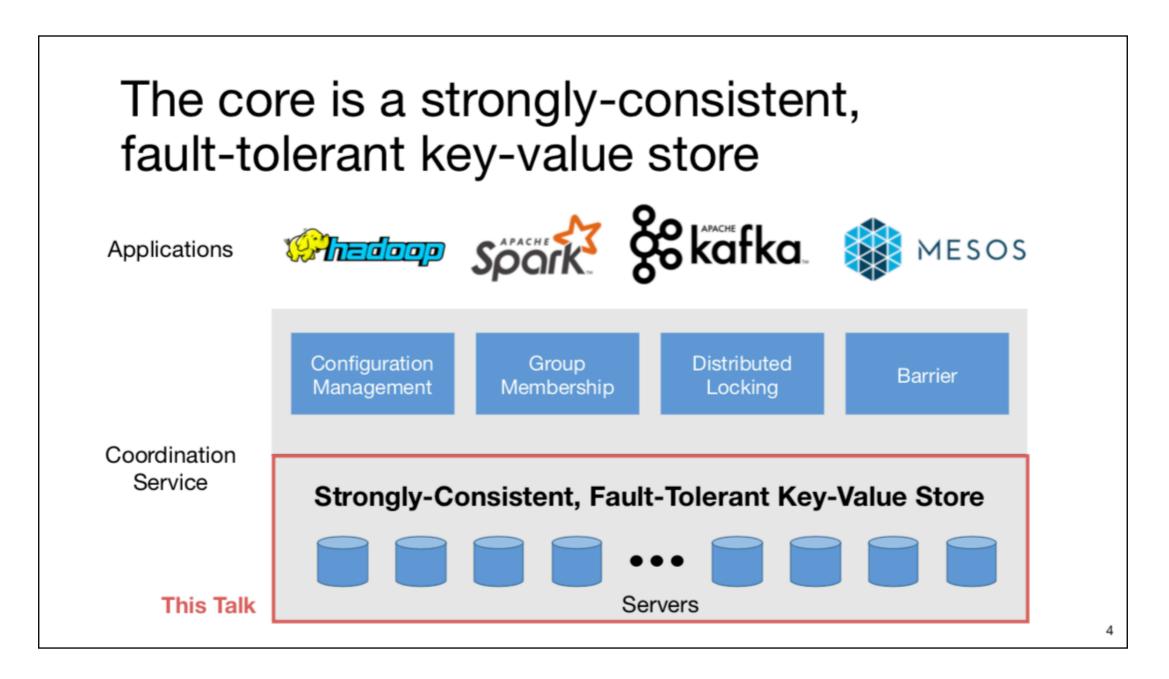
NetChain builds upon NetCache to scale coordination services, a key building block of distributed systems

Conventional wisdom: avoid coordination

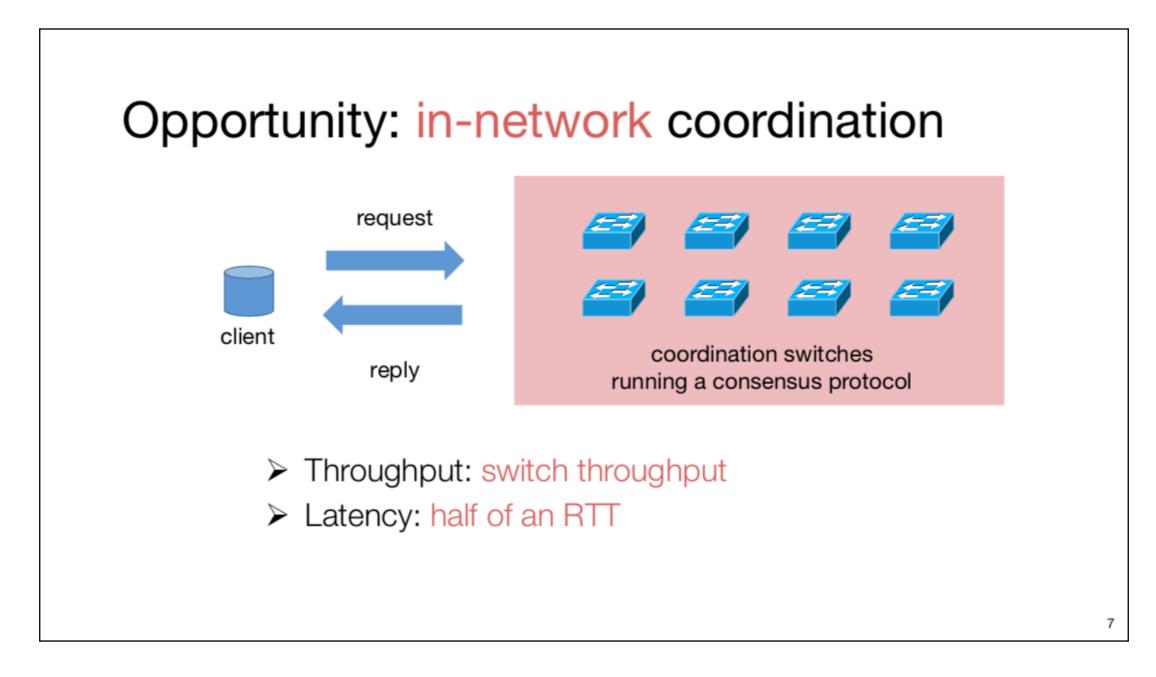
NetChain: lightning fast coordination enabled by programmable switches

Open the door to rethink distributed systems design

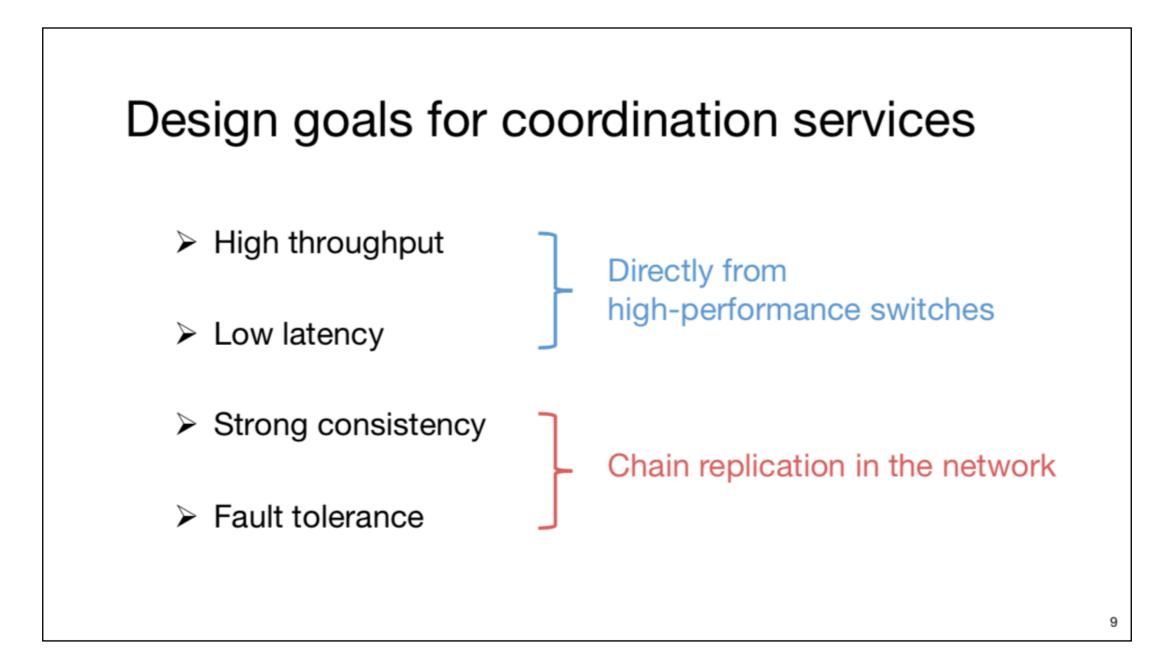
Coordination services typically rely on a replicated key-value store for consistency and fault-tolerance



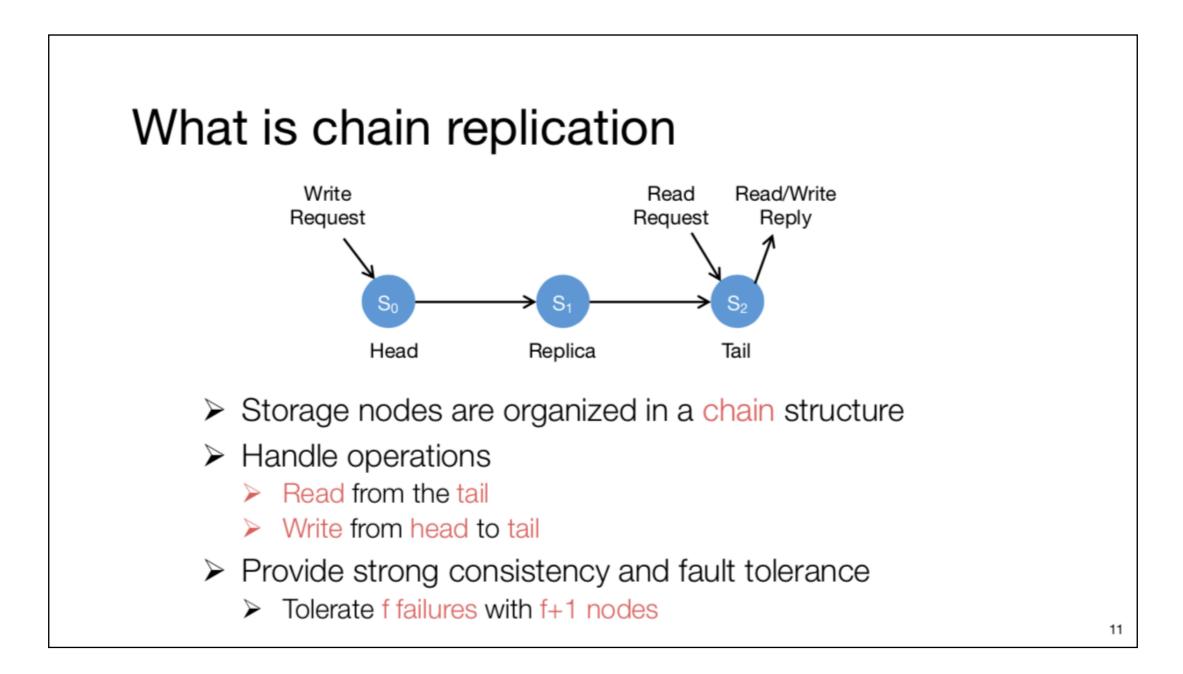
State of the art server-based coordination services struggle to provide high-throughput and low-latency



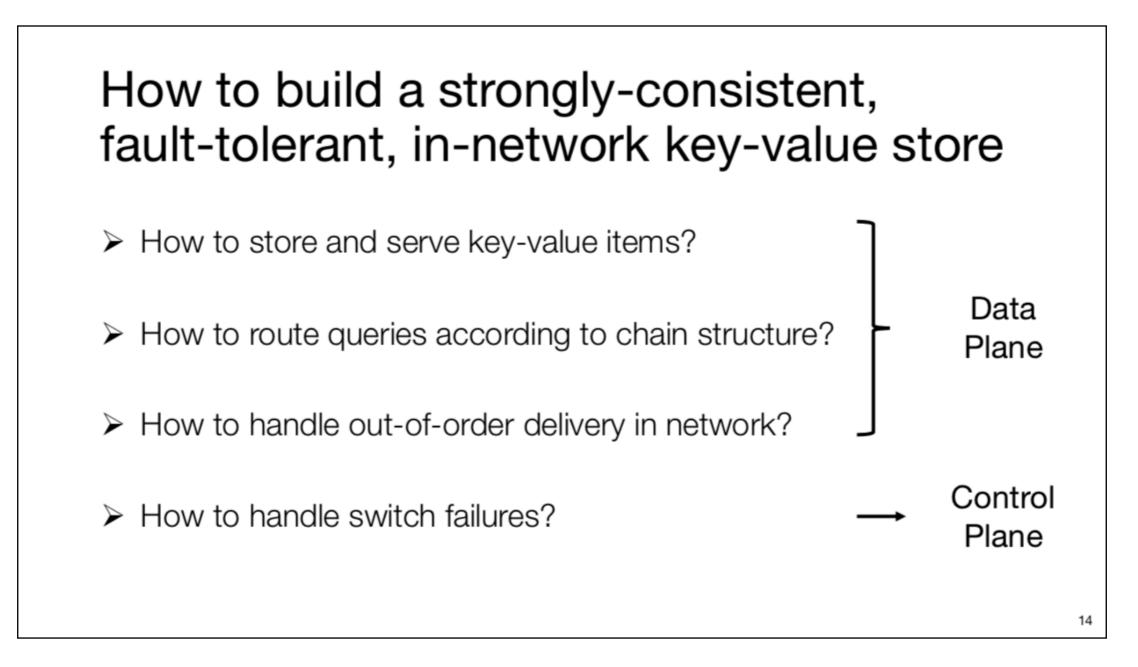
#### Key challenge is to ensure consistency and fault-tolerance



NetChain does so using chain replication, building upon NetCache for storing values in each switch



NetChain relies on a tailored UDP-based protocol, source-routing mechanisms and message serialization



### This week on

### Advanced Topics in Communication Networks

A high-level, non-exhaustive overview of the research surrounding data plane programmability A high-level, non-exhaustive overview of the research surrounding data plane programmability

Data plane for programmability

Performance Monitoring Applications offloading

PlatformsforData planeCorrectnessprogrammabilityManagement

Data plane for programmability

#### Performance

Monitoring Applications offloading

Platforms Correctness Management for Data plane programmability

# A large set of papers on programmable data planes aim at improving performance, esp. load balancing

ୁ conga-sigcomm ୧. 🍈	4.pdf (page 1 of 12)				
			le Load Balancing Using mable Data Planes		
		riogram			
	tion Among Logal Delensing		*Deinseten Uleinereite	ghoon Kimi, Anirudh Sivaraman, Jennifer Rexford	
CONGA: Distributed Congestion-Aware Load Balancing for Datacenters		{nkatta, jre		> ghorbanitTotil.pdf (page 1 of 14) ✓ → ② ③ Q. Search	
		ABSTRACI			
Mohammad Alizadeh, Tom Edsall, Sarang Dha	rmapurikar, Ramanan Vaidvanathan, Kevin Chu.	Datacenter netwy Spine, Fai-Tree) I			DRILL [SIGCOM
Andy Fingerhut, Vinh The Lam (Google), Francis Matus, Rong Pan, Navindra Yadav,		gies use a large di balancing mecha		DDII I - Misse I - ed Delan sina (en	DRIEL [SIGCOM
George Varghese (Microsoft)		width. The canor path routing (EC		DRILL: Micro Load Balancing for Low-latency Data Center Networks	
Cisco	Systems	tiple paths. Motiv load-balancing te		,	
		These techniques ory is limited, the	Soudeh University of W	isconsin-Madison University of Illinois at University of Illinois at	
ABSTRACT	to paths, hash collisions can cause significant imbalance if there are a few large flows. More importantly, ECMP uses a purely <i>local</i> de-	tracking state at t gies. Second, be	••	●	earch
We present the design, implementation, and evaluation of CONGA, a network-based distributed congestion-aware load balancing mech-	cision to split traffic among equal cost paths without knowledge of	they cannot be m This paper pre			
anism for datacenters. CONGA exploits recent trends including	potential downstream congestion on each path. Thus ECMP fares poorly with asymmetry caused by link failures that occur frequently	rithm that overco the leaf switches	ABSTRACT		
the use of regular Clos topologies and overlays for network vir- tualization. It splits TCP flows into flowlets, estimates real-time	and are disruptive in datacenters [17, 34]. For instance, the recent	each HULA switt tion through a ne	The trend towards		
congestion on fabric paths, and allocates flowlets to paths based	study by Gill et al. [17] shows that failures can reduce delivered traffic by up to 40% despite built-in redundancy.	emerging program strate that HUL#	work functionality and pushes it to t		
on feedback from remote switches. This enables CONGA to effi-	Broadly speaking, the prior work on addressing ECMP's short-	without requiring sively in simulati	main culprit of pac direction: could s		
ciently balance load and seamlessly handle asymmetry, without re- quiring any TCP modifications. CONGA has been implemented in	comings can be classified as either centralized scheduling (e.g.,	sion to CONGA load, 3× at 90%	balancing? This p networks which p	Let it Flow: Resilient Asymmetric Load Balancing with Flowlet Switchi	ing
custom ASICs as part of a new datacenter fabric. In testbed exper-	Hedera [2]), local switch mechanisms (e.g., Flare [27]), or host- based transport protocols (e.g., MPTCP [41]). These approaches		evenly as possible packet decisions :	Erico Vanini* Rong Pan* Mohammad Alizadeh <sup>†</sup> Parvin Taheri* Tom Eds	all*
iments, CONGA has 5× better flow completion times than ECMP even with a single link failure and achieves 2-8× better through-	all have important drawbacks. Centralized schemes are too slow	CCS Concep	and randomized a the resulting key	*Cisco Systems *Massachusetts Institute of Technology	
put than MPTCP in Incast scenarios. Further, the Price of Anar-	for the traffic volatility in datacenters [28, 8] and local congestion-	$\bullet Networks \to P_1$	asymmetry. In sir		
chy for CONGA is provably small in Leaf-Spine topologies; hence	aware mechanisms are suboptimal and can perform even worse than ECMP with asymmetry (§2.4). Host-based methods such as		and realistic work balancers, particu	Abstract better load balancing designs for datacenter netwo	
CONGA is nearly as effective as a centralized scheduler while be- ing able to react to congestion in microseconds. Our main thesis	MPTCP are challenging to deploy because network operators often	Keywords	ample, it achieves recent proposals,	Datacenter networks require efficient multi-path load A defining feature of these designs is the inf tion that they use to make decisions. At one end	
is that datacenter fabric load balancing is best done in the network,	do not control the end-host stack (e.g., in a public cloud) and even when they do, some high performance applications (such as low	In-Network Loac Congestion; Scal	hardware feasibili its area overhead	balancing to achieve high bisection bandwidth. Despite much progress in recent years towards addressing this	
and requires global schemes such as CONGA to handle asymmetry. Categories and Subject Descriptors: C.2.1 [Computer-Communication	latency storage systems [39, 7]) bypass the kernel and implement	Congestion, other	stability and throe	challenge, a load balancing design that is both simple to implement and resilient to network asymmetry has	o [15],
Networks]: Network Architecture and Design	their own transport. Further, host-based load balancing adds more complexity to an already complex transport layer burdened by new	Permission to make d classroom see is grant	CCS CONCE • Networks Netv	remained elusive. In this paper, we show that flowlet maintain an idea first proposed more than a decade and chunks of data (called "flowcells"), fall in this cat	iegory.
Keywords: Datacenter fabric; Load balancing; Distributed	requirements such as low latency and burst tolerance [4] in data-	for profit or comment tion on the first page. ACM must be bonners	KEYWORDS	is a powerful technique for resilient load balancing with	
1. INTRODUCTION	centers. As our experiments with MPTCP show, this can make for	publish, to post on ser and/or a fee. Request	Microbursts, Load	asymmetry. Flowlets have a remarkable <i>elasticity</i> prop- erty: their size changes automatically based on traffic bes are CONGA [3] and HULA [21], which use	
Datacenter networks being deployed by cloud providers as well	brittle performance (§5). Thus from a philosophical standpoint it is worth asking: Can	5058/16, March 14- © 2016 ACM, ISBN	ACM Reference for Soudeh Ghorbani,	conditions on their path. We use this insight to develop back between the switches to gather path-wise or	onges-
as enterprises must provide large bisection bandwidth to support	load balancing be done in the network without adding to the com-	DOE http://dx.doi.o	and Amin Firocesh latency Data Center les, CA, USA, Augus	silient to asymmetry. LetFlow simply picks paths at ran- Load balancing schemes that require path cong	estion
an ever increasing array of applications, ranging from financial ser- vices to big-data analytics. They also must provide agility, enabling	plexity of the transport layer? Can such a network-based approach compute globally optimal allocations, and yet be implementable in		les, CA, USA, Augut http://dx.doi.org/10,	ance the traffic on different paths. Our extensive eval-	
vices to big-data analytics. They also must provide againty, enabling any application to be deployed at any server, in order to realize	a realizable and distributed fashion to allow rapid reaction in mi-			uation with real hardware and packet-level simulations shows that LetFlow is very effective. Despite being much	trivial
operational efficiency and reduce costs. Seminal papers such as	croseconds? Can such a mechanism be deployed today using stan- dard encapsulation formats? We seek to answer these questions		"Work done while the	simpler, it performs significantly better than other traffic 30], to implement end-to-end or hop-by-hop feed	dback.
VL2 [18] and Portland [1] showed how to achieve this with Clos topologies, Equal Cost MultiPath (ECMP) load balancing, and the	in this paper with a new scheme called CONGA (for <i>Congestion</i>		Permission to make di	ric scenarios, while achieving average flow completions	orly in
decoupling of endpoint addresses from their location. These de-	Aware Balancing). CONGA has been implemented in custom ASICs		classroom use is grant for profit or commercia on the first page. Cop	and 2× of CONGA in simulated topologies with large	
sign principles are followed by next generation overlay technolo- gies that accomplish the same goals using standard encapsulations	for a major new datacenter fabric product line. While we report on lab experiments using working hardware together with simulations		author(s) must be hose republish, to post on se and/or a for. Roquest p	asymmetry and heavy traffic load. reason is tinat the optimat trainciplit across asy rice paths depends on (dynamically varying) traffi ditions: hence, traffic-oblivious schemes are fund	c con-
such as VXLAN [35] and NVGRE [45].	and mathematical analysis, customer trials are scheduled in a few			1 Introduction tally unable to make optimal decisions and can pe	
However, it is well known [2, 41, 9, 27, 44, 10] that ECMP can belance load peoply. First because ECMP randomly backet flows	months as of the time of this writing. Figure 1 surveys the design space for load balancing and places		0.2017 Copyright held ACM 978-1-4503-465 http://dx.doi.org/10.11	Datacenter networks must provide large bisection hand. Asymmetry is common in practice for a variety of	of rea-
balance load poorly. First, because ECMP randomly hashes flows	CONGA in context by following the thick red lines through the de-			width to support the increasing traffic demands of ap- plications such as big-data analytics, web services, and equipment [31, 12, 3]. Handling asymmetry grac	twork
Permission to make digital or hard copies of all or part of this work for personal or	sign tree. At the highest level, CONGA is a distributed scheme to				
classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full cita-	allow rapid round-trip timescale reaction to congestion to cope with bursty datacenter traffic [28, 8]. CONGA is implemented within the			over many paths in multi-rooted tree topologies such as Clos [13] and Fat-tree [1]. These designs are widely de-	tion in
tion on the first page. Copyrights for components of this work owned by others than	network to avoid the deployment issues of host-based methods and			ployed: for instance, Google has reported on using Clos fabrics with more than 1 Pbps of bisection bandwidth in	cheme
ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or re- publish, to post on servers or to redistribute to lists, requires prior specific permission	additional complexity in the transport layer. To deal with asymme- try, unlike earlier proposals such as Flare [27] and LocalFlow [44]			its datacenters [25]. and yet it is very resilient to network asymmetry.	
and/or a fee. Request permissions from permissions@acm.org. SIGCOMM/14, August 17-22, 2014. Chicago, IL, USA.	that only use local information, CONGA uses global congestion			contract. Equal Cost MultiPath (ECMP) 1161 and only and on for each flowlet. That's it! A flowlet is a	a burst
Copyright 2014 ACM 978-1-4503-2836-4/14/08\$15.00.	information, a design choice justified in detail in §2.4.			assigns flows to different paths using a hash taken over nacket headers. ECMP is widely denloved due to its sim-	rsts by Jowlet
http://dx.doi.org/10.1145/2619239.2626316 .				plicity but suffers from well-known performance prob- lems such as hash collisions and the inability to adapt	ago as
				lems such as hash collisions and the inability to adapt to asymmetry in the network topology. A rich body of work [10, 2, 22, 23, 18, 3, 15, 21] has thus emerged on	overin

CONGA [SIGCOMM'14]

HULA [SOSR'16]



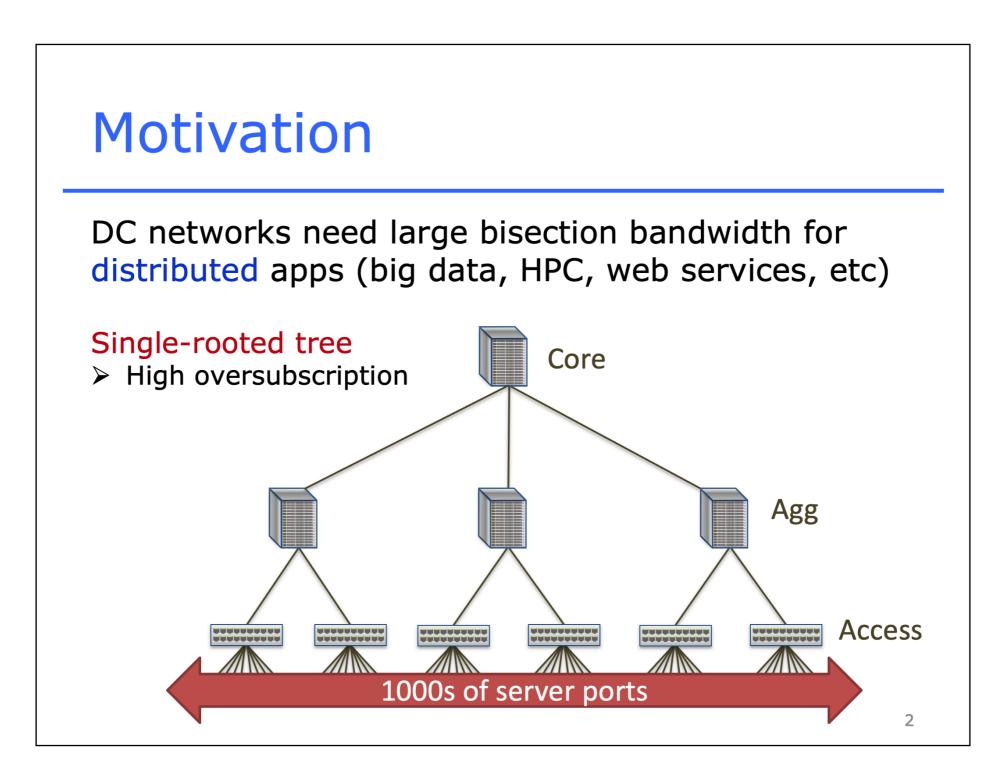
# A large set of papers on programmable data planes aim at improving performance, esp. load balancing

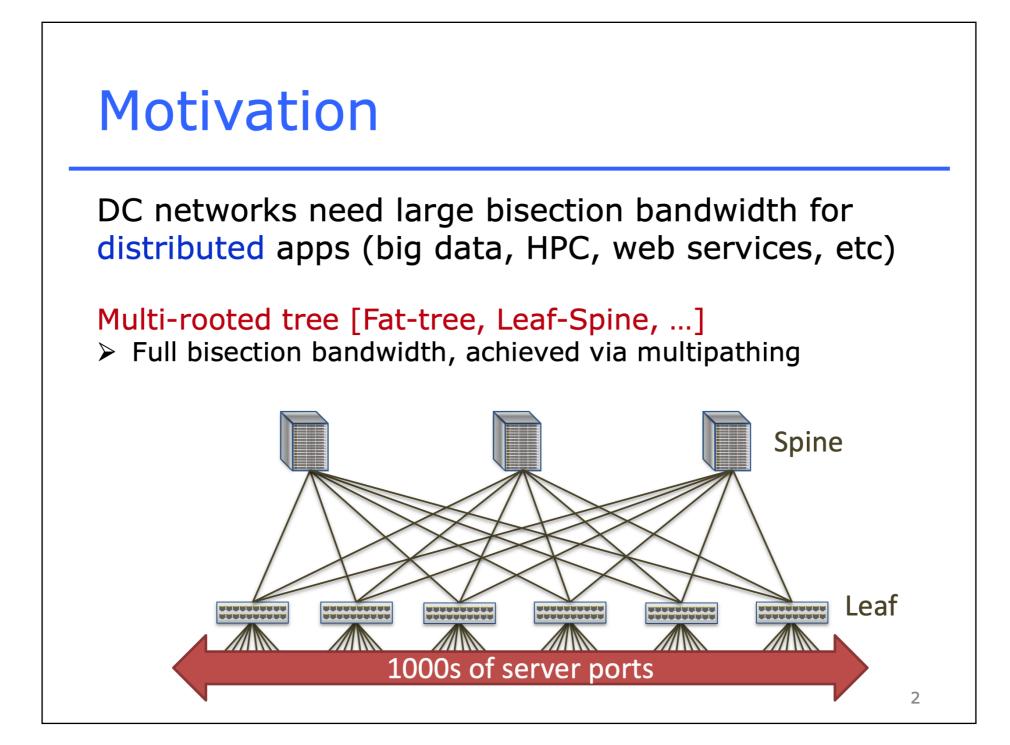
CONGA [SIGCOMM'14]

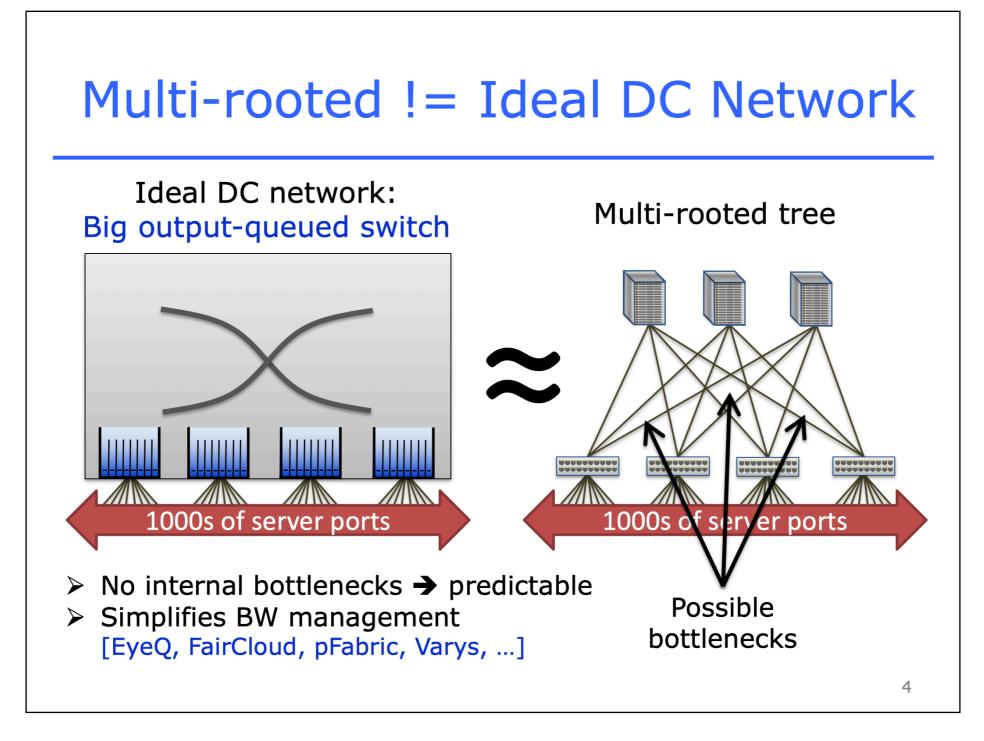
2 0 0	a conga-sigcomm14.pdf (page 1 of 12) ✓ ▷ ② Q. Search				
				ble Load Balancing Using	
			Program	nmable Data Planes	
	ributed Congestion-Aware Load Balancing			nghoon Kimi, Anirudh Sivaramani, Jennifer Rexford	
CONGA. DISI		(nkatta, jre			
	for Datacenters				
		ABSTRACT			
Male array and Alian date	Terr Educil Conner Discoversition Demons Weidemother Kerin Obu				DRILL [SIGCOMM'
	Tom Edsall, Sarang Dharmapurikar, Ramanan Vaidyanathan, Kevin Chu, /inh The Lam (Google), Francis Matus, Rong Pan, Navindra Yadav,				
Andy Fingemat,	George Varghese (Microsoft)			DRILL: Micro Load Balancing for	
	Cisco Systems			Low-latency Data Center Networks	
	Claco Gyatema			Ghorbani" Zibin Yang P. Brighten Godfrey	
ABSTRACT	to paths, hash collisions can cause significant imbalance if there are			Visconsin-Madison University of Illinois at	
We present the design, implementa	a few large flows. More importantly, ECMP uses a purely local de-				
a network-based distributed conges	tion-aware load balancing mech-				
anism for datacenters. CONGA the use of regular Clos topologie	Aprons recent using the second by the follower dust second by		ABSTRACT		
tualization. It splits TCP flows in	and overlays for network vir-				
congestion on fabric paths, and a			and pushes it to t main culprit of ps		
on feedback from remote switches ciently balance load and seamlessl	bardle scurmmatry without ra				
quiring any TCP modifications. Co				Let it Flow: Resilient Asymmetric Load Balancing with Flowlet S	Switching
custom ASICs as part of a new dat iments, CONGA has 5× better flo	second at than ECMb based transport protocols (e.g., MPTCP [41]). These approaches	CCS Concer		Erico Vanini* Rong Pan* Mohammad Alizadeh <sup>†</sup> Parvin Taheri* T	lom Edsall"
even with a single link failure and	achieves 2–8× better through- all have important drawbacks. Centralized schemes are too slow		and randomized a the resulting key	*Cisco Systems <sup>†</sup> Massachusetts Institute of Technology	
put than MPTCP in Incast scenar chy for CONGA is provably small	aware mechanisms are suboptimal and can perform even worse	$\bullet Networks \to P$	asymmetry. In sir and realistic work		
CONGA is nearly as effective as a	centralized scheduler while be- than ECMP with asymmetry (§2.4). Host-based methods such as	Keywords		Abstract better load balancing designs for datacer A defining feature of these designs	
ing able to react to congestion in is that datacenter fabric load balan				Datacenter networks require efficient multi-path load balancing to achieve high bisection bandwidth. Despite much pureness in recent versar towards addressing that are obliviou	t one end of the
and requires global schemes such a	CONGA to handle asymmetry when they do, some high performance applications (such as low				
Categories and Subject Descriptors	C.2.1 [Computer-Communication] Latency storage systems [39, 7]) bypass the kernel and implement their own transport. Further, host-based load balancing adds more		CCS CONCE	to implement and resilient to network asymmetry has remained elusive. In this paper, we show that flowlet	
Networks]: Network Architecture and Keywords: Datacenter fabric; Load ba	Design complexity to an already complex transport layer burdened by new		• Networks Netv		
	requirements such as low latency and burst tolerance [4] in data- centers. As our experiments with MPTCP show, this can make for		KEYWORDS		
1. INTRODUCTION	brittle performance (§5).		Microbursts, Loac ACM Reference for		
Datacenter networks being depl			ACM Reference fo Soudeh Ghorbani, and Amin Firocesh	LetFlow, a very simple load balancing scheme that is re- silient to asymmetry. LetFlow simply picks paths at ran- dom for floatenet and hit traffic a lustricity articular block.	
as enterprises must provide large an ever increasing array of applicat					
vices to big-data analytics. They al	o must provide agility, enabling compute globally optimal allocations, and yet be implementable in		http://dx.doi.org/10.	uation with real hardware and nacket-level simulations	
any application to be deployed at operational efficiency and reduce				simular in parforms similar and better than other traffic	
VL2 [18] and Portland [1] showed	how to achieve this with Clos dard encapsulation formats? We seek to answer these questions			objivious schemes like WCMP and Presto in asymmet- ric scenarios, while achieving average flow completions time within 10-20% of CONR6 has testbed exerciments	ability into path
topologies, Equal Cost MultiPath decoupling of endpoint addresses	ECMP) load balancing, and the from their location. These de- Aware Balancing). CONGA has been implemented in custom ASICs				
sign principles are followed by ne	xt generation overlay technolo- for a major new datacenter fabric product line. While we report on			and 2x of CONGA in simulated topologies with large asymmetry and heavy traffic load.	
gies that accomplish the same goa such as VXLAN [35] and NVGRE				ditions; hence, traffic-oblivious schemes	
However, it is well known [2, 4	, 9, 27, 44, 10] that ECMP can months as of the time of this writing.			poorly in asymmetric topologies.	
balance load poorly. First, because				Datacenter networks must provide large bisection band- width to support the increasing traffic demands of ap-	
Developing to make dising on the	sign tree. At the highest level CONGA is a distributed scheme to			plications such as big-data analytics, web services, and cloud storage. They achieve this by load balancing traffic	
Permission to make digital or hard copies or classroom use is granted without fee provide	all or part of this work for personal or d that copies are not made or distributed allow rapid round-trip timescale reaction to congestion to cope with				
for profit or commercial advantage and that tion on the first page. Copyrights for comp				Clos [13] and Fat-tree [1]. These designs are widely de- ployed; for instance, Google has reported on using Clos	
ACM must be honored. Abstracting with cre	tit is permitted. To copy otherwise, or re-			fabrics with more than 1 Pbps of bisection bandwidth in its datacenters [25]. that requires no state to make load balar and yet it is very resilient to network asy	ncing decisions, mmetry.
publish, to post on servers or to redistribute and/or a fee. Request permissions from perm	issions@acmorg. that only use local information CONCA uses aloal approximation				s pick a path at
SIGCOMM'14, August 17-22, 2014, Chicag Copyright 2014 ACM 978-1-4503-2836-4/1	0, IL, USA.			centers, Equal Cost MultiPath (ECMP) [16], randomly assigns flows to different paths using a hash taken over	
http://dx.doi.org/10.1145/2619239.2626316					
				lems such as hash collisions and the inability to adapt to asymmetry in the network topology. A rich body of	

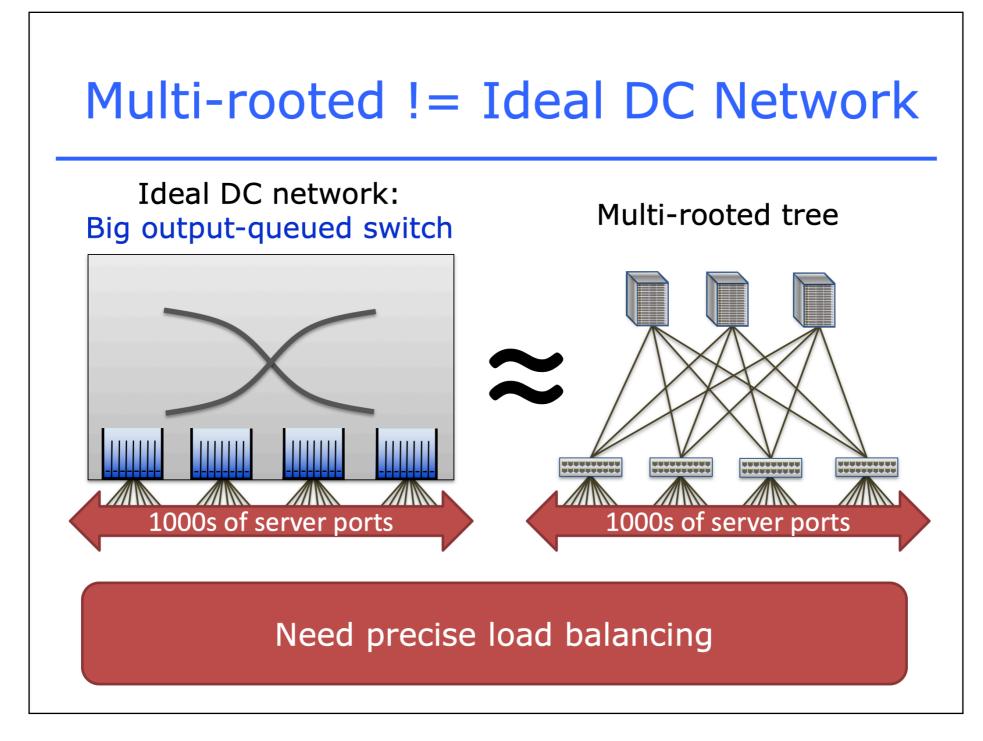
HULA [SOSR'16]

LetFlow [NSDI'17]









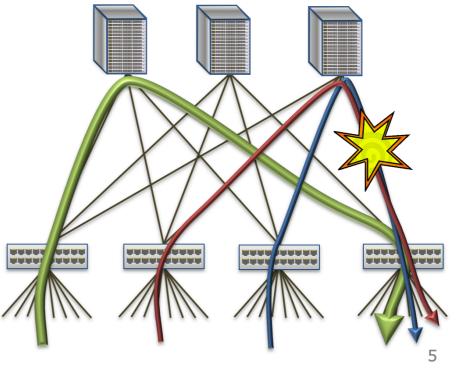
# Today: ECMP Load Balancing

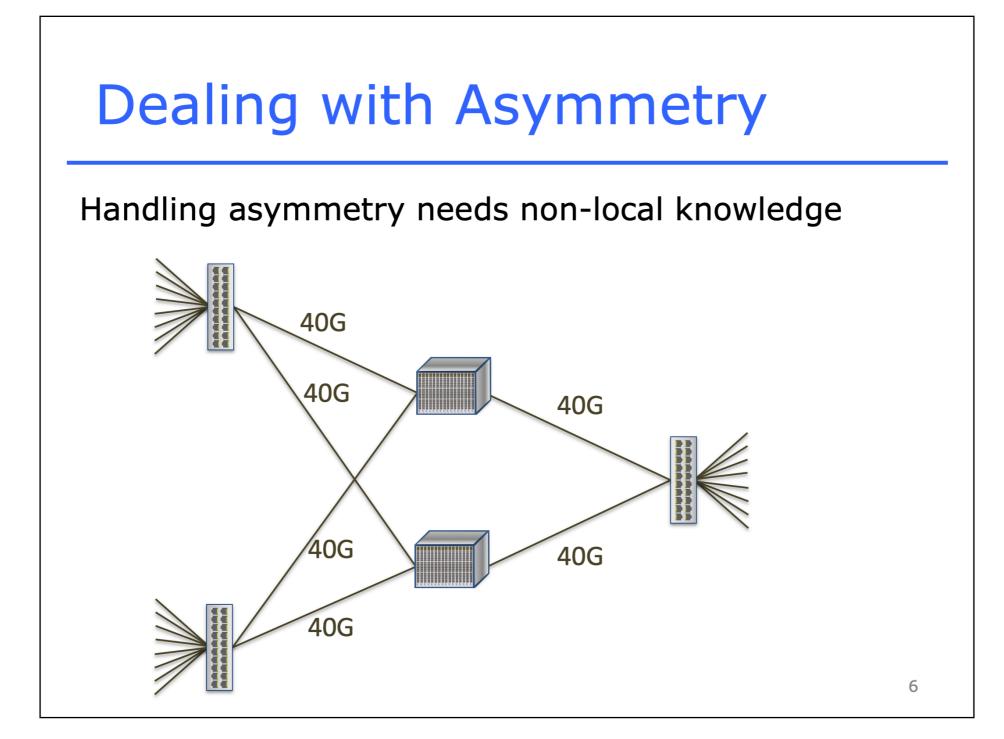
# Pick among equal-cost paths by a hash of 5-tuple➤ Approximates Valiant load balancing

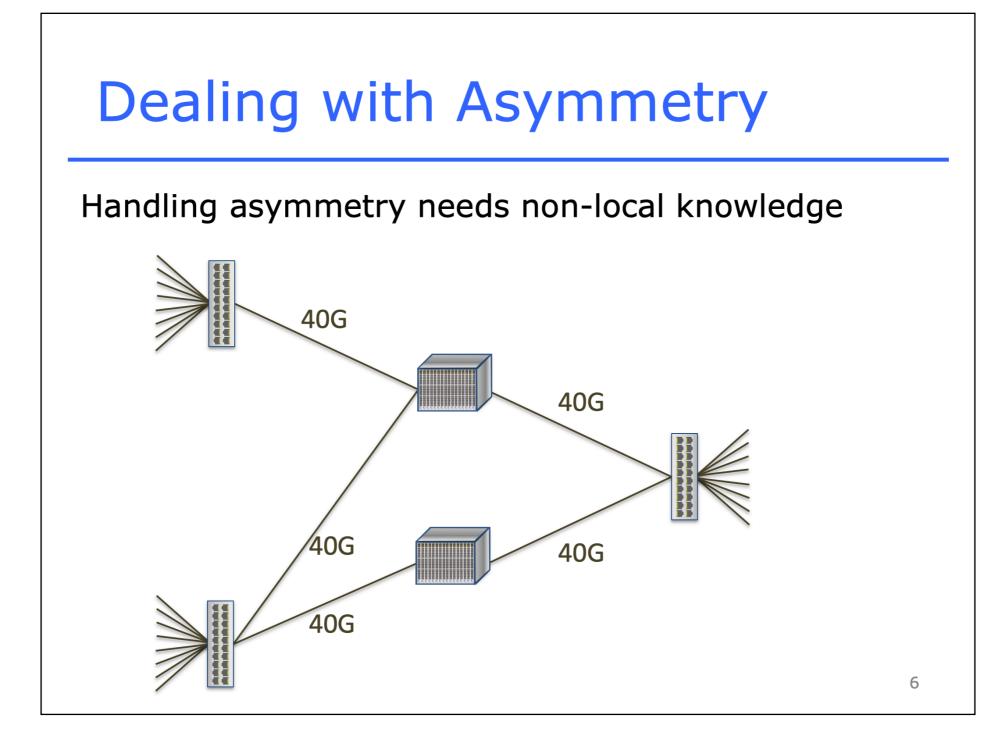
Preserves packet order

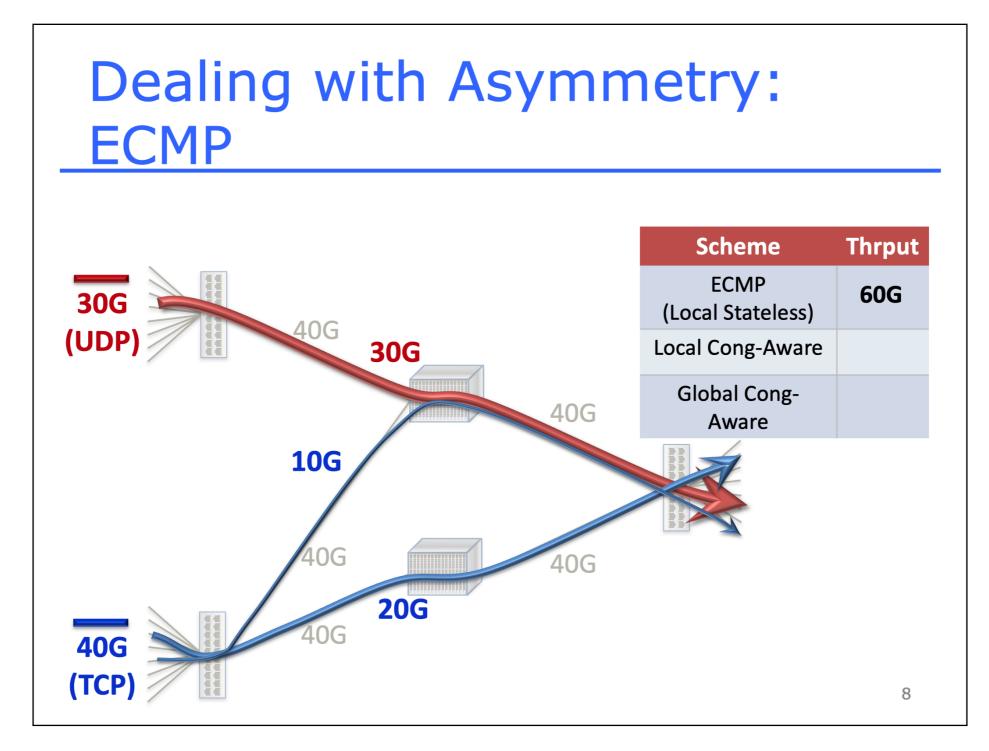
#### Problems:

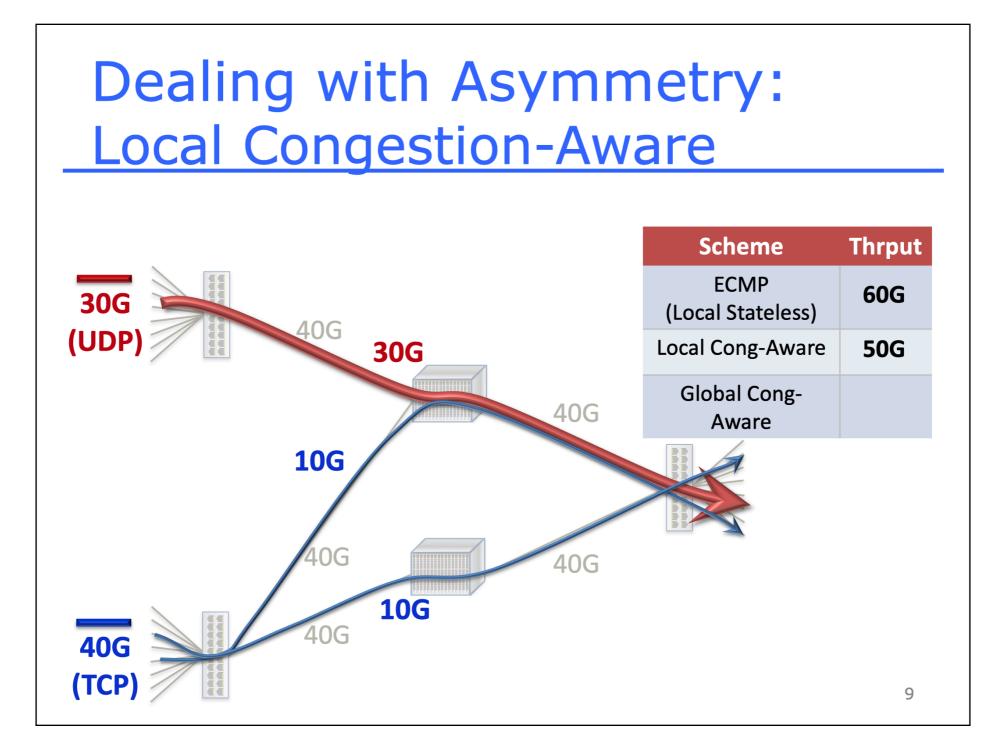
- Hash collisions (coarse granularity)
- Local & stateless
   (v. bad with asymmetry due to link failures)

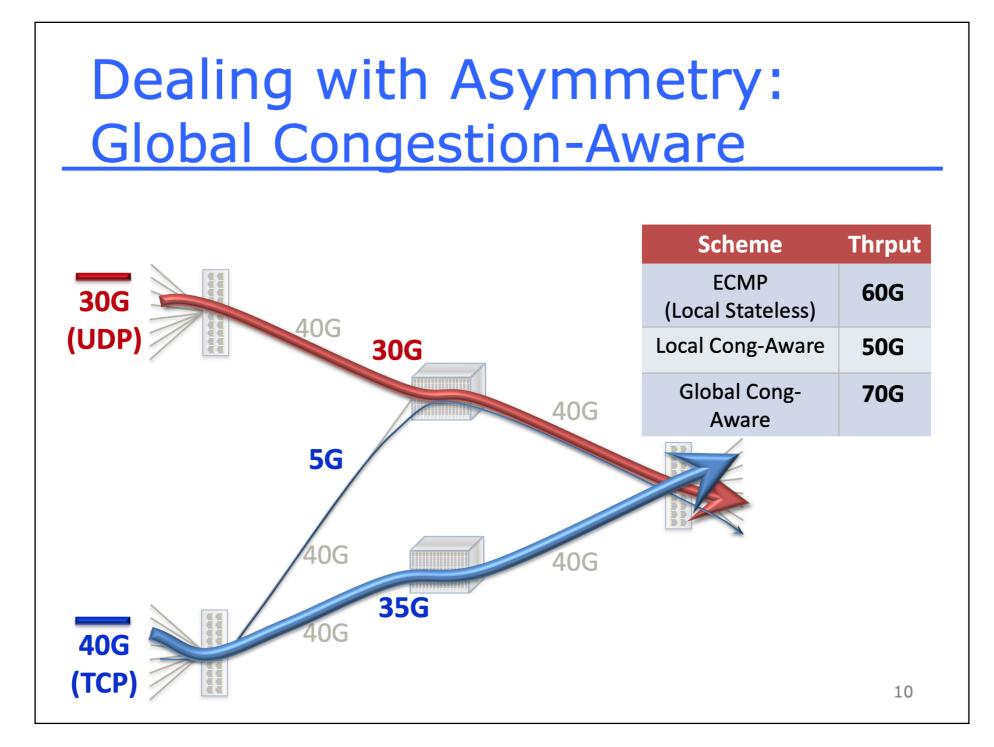


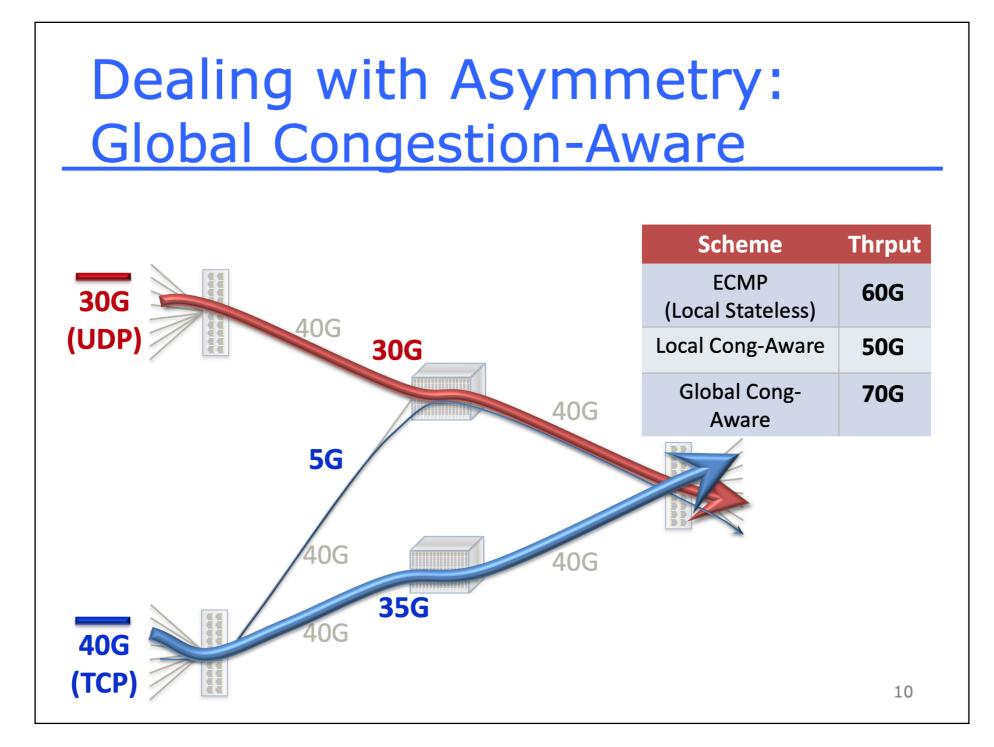


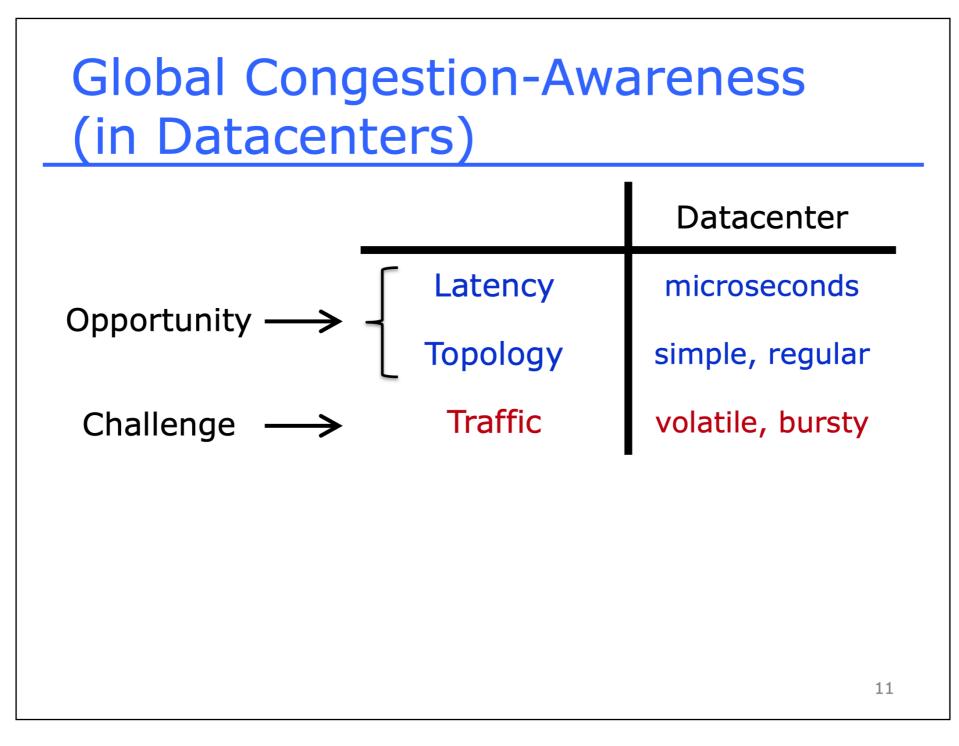


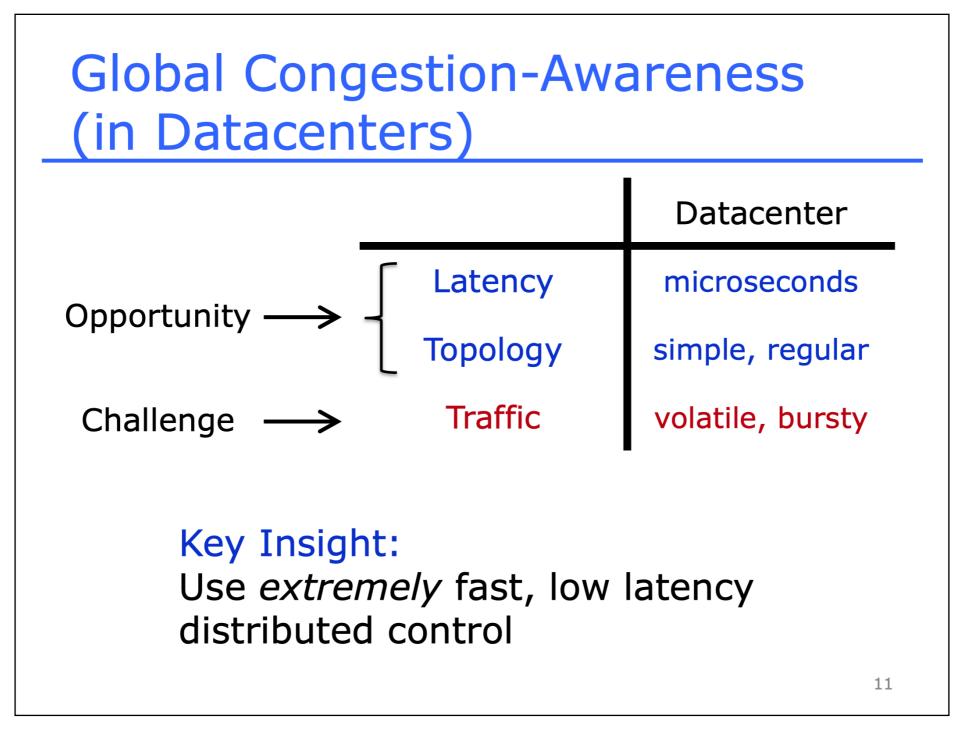


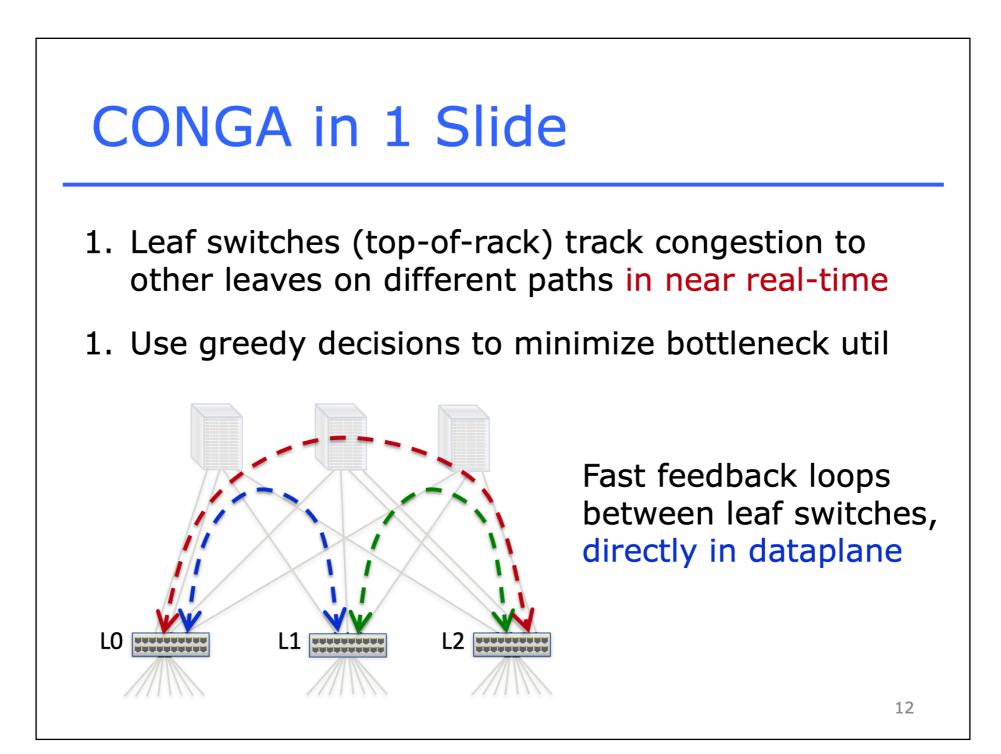




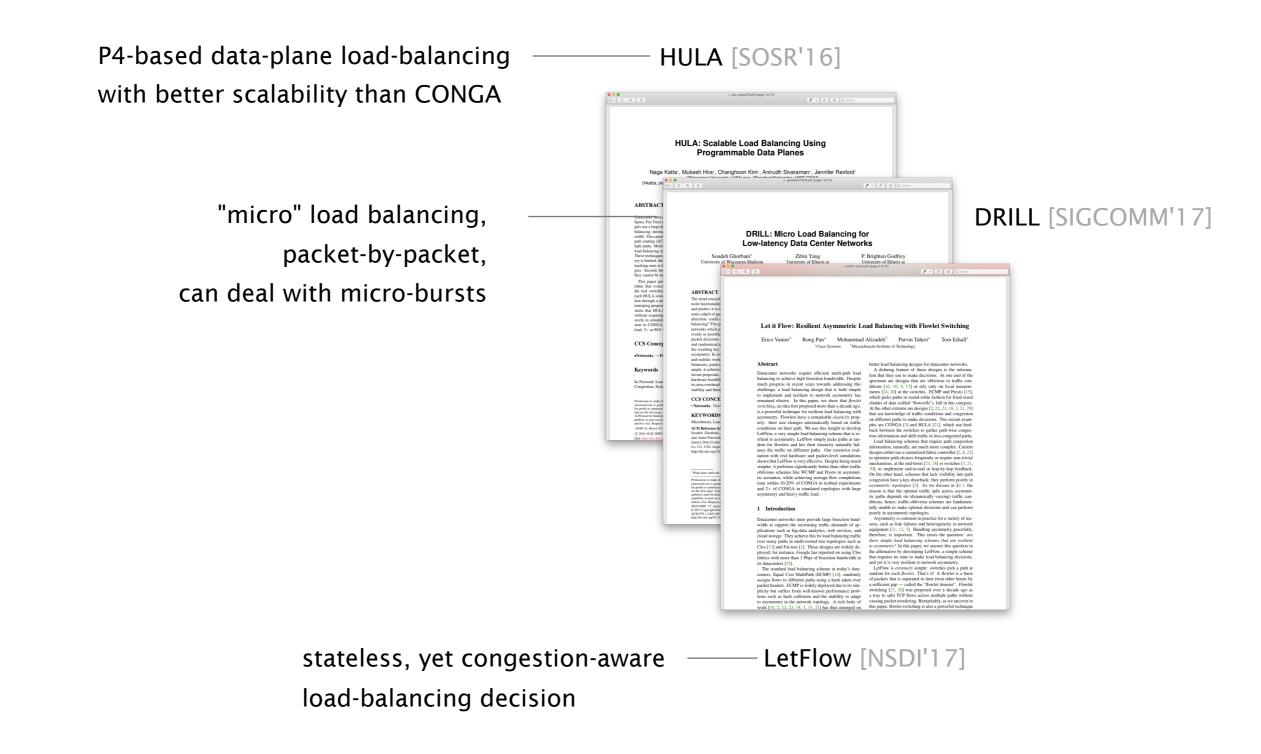








A large set of papers on programmable data planes aim at improving performance, esp. load balancing



Data plane for programmability

Performance Monitoring Applications offloading

Platforms Correctness Management for Data plane programmability

2. (1)			2 8 Q teen	
	Language-Directed Network Perforn	nance Monitoring	or	
		sonata.pdf (page 1 of 15)	🗶 💌 👌 🛞 Q Search	
AB				
Netw				
swite endp				
have				
the c given	Sonata: Quary Dr	ivon Strooming	Network Telemetry	
gives we as	Jonata: Query-Dr	iven streaming	verwork relementy	
expre that i	Arpit Gupta	Rob Harrison	Marco Canini	
varie	Princeton University	Princeton University	KAUST	
w				
is ba	Nick Feamster Princeton University	Jennifer Rexford Princeton University	Walter Willinger NIKSUN Inc.	
swite	,	,		
at lin and s	ABSTRACT	1 INTRO		
targe	Managing and securing networks requir analyzing network traffic data in real time		ators routinely perform continuous monitor- ents ranging from performance impairments	
speer that s	try systems do not allow operators to exp		is monitoring requires continuous, real-time	
only	queries needed to perform management or		and analysis-a process commonly referred	
CC	fic volumes and rates. We present Sonata, scalable telemetry system that coordinate		elemetry [55]. Existing telemetry systems can alyze measurement data in real time, but they	
• Net	and analysis of network traffic. Sonata pro-		a limited set of telemetry tasks [34, 40], or	
	interface to express queries for a wide r		stantial processing and storage costs as traffic	
KE'	telemetry tasks; to enable real-time execu tions each query across the stream proce		ies increase [7, 10, 58]. emetry systems typically trade off scalabil-	
Netw	plane, running as much of the query as it c	an on the network ity for expres	siveness, or vice versa. Telemetry systems	
ACM	switch, at line rate. To optimize the use memory, Sonata dynamically refines each		ream processors alone are expressive but not example, systems such as NetQRE [58] and	
Sriniv Venka	that available resources focus only on traffi		cample, systems such as NetQRE [58] and can support a wide range of queries using	
Kim.	query. Our evaluation shows that Sonata c	an support a wide stream proces	sors running on general-purpose CPUs, but	
Monii 21-25	range of telemetry tasks while reducing th stream processor by as much as seven or		stantial bandwidth and processing costs to do rorks can require performing as many as 100	
https:/	compared to existing telemetry systems.	million operati	ons per second for rates of 1 Tbps and packet	
	CCS CONCEPTS	sizes of 1 KB. S	caling to these rates using modern stream pro-	
-	<ul> <li>Networks → Network monitoring;</li> </ul>		ibitively costly due to the lower (2–3 orders of ocessing capacity per core [37, 39, 41, 59]. On	
Permi classre for pro		the other hand	telemetry systems that rely on programmable	
on the	KEYWORDS		e can scale to high traffic rates, but they give ness to achieve this scalability. For example,	
outhor copubli and/or	analytics, programmable switches, stream		ness to achieve this scalability. For example, nd OpenSketch [56], can perform telemetry	
SIGCO	ACM Reference Format: Arpit Gupta, Rob Harrison, Marco Canini, Nicl	tasks by execu	iting queries solely in the data plane at line	
© 2011 for Co ACM	Rexford, and Walter Willinger. 2018. Sonata: Qu	ery-Driven Stream-	eries that they can support are limited by the d memory in the data plane.	
ACM1 https:/	ing Network Telemetry. In SIGCOMM '18: Ad	CM SIGCOMM 2018 Rother than	a memory in the data plane. accepting this apparent tradeoff between ex-	
	Conference, August 20–25, 2018, Budapest, Hungo NY, USA, 15 pages. https://doi.org/10.1145/323	0543 3230555 pressiveness a	nd scalability, we observe that stream proces-	
		sors and progra model: they b	ammable switches share a common processing oth apply an ordered set of transformations	
	Permission to make digital or hard copies of all or personal or classroom use is granted without fee provi	part of this work for ded that expire are part	d data in a pipeline. This commonality sug-	
	made or distributed for profit or commercial advantage	the that copies are not gests that an o	pportunity exists to combine the strengths of	
	this notice and the full citation on the first page. Copy of this work owned by others than ACM must be hen		ies in a single telemetry system that supports ries, while still operating at line rate for high	
	credit is permitted. To copy otherwise, or republish, t	o post on servers or to traffic volumes	s and rates.	
	redistribute to lists, requires prior specific permission permissions from permissions@acm.org.	and/or a fee. Request To explore t	this idea, we develop Sonata (Streaming Net-	
	SIGCOMM '18, August 29-25, 2018, Budapest, Hangar © 2018 Association for Computing Machinery.		Analysis), a <i>query-driven</i> network telemetry 2 1 shows the design of Sonata: it provides	

2 (Q ()	(≤ conext16.pdf (page 1 of 15)	🗶 * 👌 🛞 Q Seech
LossRa	dar: Fast Detection of Lost P Center Networks	ackets in Data
		🗶 💌 🗇 💿 Q, Search
ABSTR. Packet loss caused by a and have sit		
and nave sq network of detection o We also nee information these losse are generic fall short in low overhe- we propose		etFlow for Data Centers hangboon Kim <sup>†</sup> Minlan Yu <sup>*</sup> fornia <sup>†</sup> Barefoot Networks
detection. V individual I entire netwo on prototyp is easy to i memory an information analysis to	Abstract NetFlow has been a widely used monitoring tool with a variety of applications. NetFlow maintains an active working set of flows in a hash hubb fut augueors flow	plement NetFlow in hardware is how to maintain an ac- tive working set of flows using a data structure with low time and space complexity. We need to handle collisions during flow insertion and remove out flows to make room
with a few of the second secon	working see of hows in a hash take the unit supports how interface, collision resolution, and flow removing. This is hard to implement in merchant silicon at data cen- ter switches, which has imitted per-packet processing time. Therefore, many NetFlow implementations and other monitoring solutions have to sample or select a subset of packets to monitor. In this paper, we observe the need to monitor all the flows without sampling in	for new ones. These tasks are challenging given the lim- ited per-packet processing inter at merchant silicon. To handle this challenge, today's NetFlow is imple- mented in two ways: (1) Using complex castom silicon that is only available at high-end routers, which is too expensive for data centers; (2) Using software to count sampled packets from hardware, which takes too much
for one yes losses, 4 n random cor immediate significant especially v	the need to monitor at the nows without sampling in short time scales. Thus, we design FlowRadar, a new way to maintain flows and their contents that scales to a large number of flows with small memory and contant inser- flow contents with a small memory and constant inser- tion time at switches, and then to leverage the computing power at the remote collector to perform network-wide	CPU resources at switches. Because of the lack of us- able NetFlow in data centers, operators have to mirror packets based on sampling or matching rules and ma- type these packets in a remost collector [25, 40, 44, 34]. It is impossible to mirror all the packets because it takes too much bandwidth to mirror the traffic, and too many storage and computing resources as the remote collector
or classroom a distributed for and the full cit owned by othe mitted. To cop lists, requires p permissions/#a <i>CoNEXT</i> 'J © 2016 ACM.	decoding and analysis of the flow counters. Our eval- uation shows that the memory usage of PowRadar is close to traditional NetFlow with <i>perfect hashing</i> . With FlowRadar, operators can get better views into their net- works as demonstrated by two new monitoring applica- tions we build on top of FlowRadar.	to analyze every packet. (Section 2) However, in duce centers, there is an increasing need to have visibility of the counters for all the flows all the time. We need to over all the flows to capture show tran- sient loops, blackholes, and witch fulls that only hap- pen to a feer flows in the Network and to perform fine- grained traffic analysis (e.g., anomaly detection). We need to over these flows all the time to detainfut transient
por here/20	I Introduction NetFlow [4] is a widely used menineing tool for over 20 years, which records the flows (e.g., source IP, domin- tion IP, source port, domination or early and the ing and finite theory. When a low finite share the in- service timeset, NetFlow exports the corresponding flow records to a sume collection. NetFlow has been used for a variety of monitoring applications such as accounting network wage, capacity planning, trobschooling, and ana.1 detection.	bases, benst, and attack in a timely fablon. (Stevins 7) In this paper, we propeo Flowkalar, which keeps counters for all the flows with low memory overhead and exposits flow four counters in short time taskles e.g., 10 ma. <sup>3</sup> The key design of Flowkalar is to identify the end propusate propositing to mark the enter collector with plotty of computing resources. We introduce en- coded flowers that we propiest imple contast time in- structions for each packet and thus are easy to implement with merchan siloan at deep switches. We thus does these does and apple at the provide siloan at deep services. We introduce these does and perform activated wide analysis across time and witches and the memory collector. We name

Q Search
oot Networks rook: Arista nwani: Dell n Daly: Intel
vie: VMware



	Language-Directed				
	Network Perform		0		
AB: Netw swite endp have the c gives we a	Sonata: Query-Dr	iven Stre	aming N	etwork Telemetry	
	Arpit Gupta				
				KAUST	
W	Nick Feamster				
is bu	Princeton University				
	ABSTRACT		1 INTRODU	ICTION	
	try systems do not allow operators to ex queries needed to perform management or fic volumes and rates. We present Sonata,		measurement an to as network tele	ionitoring requires continuous, real-time d analysis—a process commonly referred metry [55]. Existing telemetry systems can	
CC				te measurement data in real time, but they limited set of telemetry tasks [34, 40], or	
• Net					
KE'				increase [7, 10, 58]. etry systems typically trade off scalabil-	
ACM	switch, at line rate. To optimize the use				
Kim.	query. Our evaluation shows that Sonata c	an support a wide			
	range of telemetry tasks while reducing the			atial bandwidth and processing costs to do ks can require performing as many as 100	
	CCS CONCEPTS			ing to these rates using modern stream pro-	
	<ul> <li>Networks → Network monitoring:</li> </ul>				
			the other hand, tel	emetry systems that rely on programmable	
	KEYWORDS			n scale to high traffic rates, but they give s to achieve this scalability. For example,	
	analytics, programmable switches, stream				
	ACM Reference Format: Arpit Gupta, Rob Harrison, Marco Canini, Nicl				
				ies that they can support are limited by the semory in the data plane.	
				scalability, we observe that stream proces-	
				nable switches share a common processing apply an ordered set of transformations	
				in a single telemetry system that supports s, while still operating at line rate for high	
			traffic volumes ar		
				idea, we develop <i>Sonata</i> (Streaming Net- lysis), a <i>query-driven</i> network telemetry	

		and the Party of t
LOSSHad	lar: Fast Detection of Lost Pa	ackets in Data
	Center Networks	
	Ochter Networks	
ABSTR.		
ADSIN		
and have sig		
network og		
We also nee	FlowRadar: A Rottor N	etFlow for Data Centers
	riowikauai, A Dettel IV	CLARGE AVELE CONTERS
these losse		
are generic		
	Yuliang Li* Rui Miao* Cl	hanghoon Kim <sup>†</sup> Minlan Yu <sup>*</sup>
		fornia <sup>†</sup> Barefoot Networks
detection. V		
	Abstract	
		tive working set of flows using a data structure with low
memory an		time and space complexity. We need to handle collisions
information		
analysis to:		during flow insertion and remove old flows to make room
		for new ones. These tasks are challenging given the lim-
	insertion, collision resolution, and flow removing. This	
	is hard to implement in merchant silicon at data cen-	
1. INT		
		mented in two ways: (1) Using complex custom silicon
Packet loss		that is only available at high-end routers, which is too
	subset of packets to monitor. In this paper, we observe	
	the need to monitor all the flows without sampling in	
losses, 4 n		able NetFlow in data centers, operators have to mirror
random cor		
	large number of flows with small memory and bandwidth	
	overhead. The key idea of FlowRadar is to encode per-	It is impossible to mirror all the packets because it takes
		too much bandwidth to mirror the traffic, and too many
		storage and computing resources at the remote collector
	decoding and analysis of the flow counters. Our eval-	
	uation shows that the memory usage of FlowRadar is	
	close to traditional NetFlow with perfect hashing. With	to have visibility of the counters for all the flows all the
		time. We need to cover all the flows to capture those tran-
	works as demonstrated by two new monitoring applica-	
	tions we build on top of FlowRadar.	
		grained traffic analysis (e.g., anomaly detection). We
		need to cover these flows all the time to identify transient
	1 Introduction	
		In this paper, we propose FlowRadar, which keeps
	NetFlow [4] is a widely used monitoring tool for over 20	counters for all the flows with low memory overhead
	years, which records the flows (e.g., source IP, destina-	and exports the flow counters in short time scales (e.g.,
	tion IP, source port, destination port, and protocol) and	10 ms). The key design of FlowRadar is to identify the
		ited per-packet processing time and the remote collector
	active timeout, NetFlow exports the corresponding flow	with plenty of computing resources. We introduce en-
	records to a remote collector. NetFlow has been used for	coded flowsets that only require simple constant-time in-
		with merchant silicon at cheap switches. We then decode
		time and suitches all at the remote collector. We make

INT-current-spec.pdf (page 1 of 28)
V Search
In-band Network Telemetry (INT)
June 2016
Changhoon Kim, Parag Bhide, Ed Doe: <i>Barefoot Networks</i> Hugh Holbrook: <i>Arista</i> Anoop Ghanwani: <i>Dell</i> Dan Daly: <i>Intel</i> Mukesh Hira, Bruce Davie: <i>VMwar</i> e
Introduction
Terms
What To Monitor
Switch-level Information
Ingress Information
Egress Information
Buffer Information
Processing INT Headers INT Header Types
Handling INT Packets
Header Format and Location
INT over any encapsulation
On-the-fly Header Creation
Header Format
Header Location and Format INT over Geneve

Dappe	er: Data Plane F	Performance Diagn	osis of TCP	
moj				
ABSTRAC				
With more app need to diagno	Netw	ork-Wide Heavy	Hitter Detec	tion with
			ty Switches	
nts' rights to nstead, our s	Ro	b Harrison, Qizhe Cai, Arp	it Gupta, and Jennife	er Rexford
	ABSTRACT			
eting for shar r the receiver		ring tasks identify subsets of traffic	60 -	5m Interval 60s Interval
levices now o n commodity		b-k flows for a particular statistic. A witch Architecture (PISA) switch can itter" flows directly in the data plane,	(2) 40 - <sup>1</sup>	10s Interval
erformance in otype Dapper raffic. To redu			20 - 20	
			0 2 10	20 30 40 50
Keywords				ampling Rate (1/x)
Performance E			Figure 1: This graph shows between two maker ISPs [12] w	
CCS Concept work perform: wort protocols	traces. We demonstrate tect network-wide hea	• that our solution can accurately de- vy hitters with up to 70% savings in	under high sampling rates, re the monitoring interval decre	
. INTRO	communication overhe with a provable upper			
Public cloux dication perfe	1 INTRODUCT		accuracy, in order to deal v on the switches.	vith the limited resources available
		n need to identify outliers in network s or diagnose performance problems. ct unusual traffic is to perform "heavy		ocused on heavy-hitter detection c operators often need to track the s. For example, port scanners [15]
		et unusual traffic is to perform "heavy lentifies the top-k flows (or flows ex- ed threshold), according to some met-		
opies are not n ige and that coj age. Copyrigh an ACM must	ric. For example, netwo receiving traffic from a			
	high-precision in order or TCP incast [4] in re-			
		relies on analyzing packet samples ogrammable switches open up new ating traffic statistics and identifying	the chance of missing lar communication overhead	
		ating traffic statistics and identifying the data plane [17, 18, 24, 27]. These or hard conies of all or part of this work for	dinator. Additionally, networks	
			often resort to sampling defined based on the needs	
				abstantially reduced accuracy on a when traffic volumes are high. In act sampling has on accuracy while
		g User Burdens in utomated Statisti		
		g User Burdens in utomated Statist		
Measur				
Measur *State Ke *State Ke MBSTRAC vietwork me ource requi				
Measur	ement with A	utomated Statisti	r linger 1 of 10	2 2 - (2) (2) (2, mm)
Measur	ement with A		and Fast Net	2 2 - (2) (2) (2, mm)
Measur	ement with A	utomated Statisti	and Fast Net	2 2 - (2) (2) (2, mm)
Measur	ement with A	utomated Statisti	and Fast Net	2 2 - (2) (2) (2, mm)
Measur * State K USTRAC Velocity the ource requires Value appro- ource saving ure the right- shale appro- ource saving ure the right- shale appro- sources. In rations with ket the tready	Elastic :	utomated Statisti	and Fast Net	2 2 - (2) (2) (2, mm)
Measure * 'state Ke BSTRAC Setwork: me ource requir Valie appro- ource aving ne the right indense Tel cource requir Valie appro- ource aving ne the right indense Tel cource requir Valie appro- ource aving indense Tel indense Tel in	Elastic S	Sketch: Adaptive Measur	and Fast Net ements	work-wide
Measure *State Ko *State Ko ABSTRAC Sebuark me to any any any any any any sebuark me to any any any any any any any any sebuark me any any any any any any any any any any	Elastic S	utomated Statisti Sketch: Adaptive Measur	and Fast Net ements	work-wide
Measure ************************************	Elastic :	utomated Statisti Sketch: Adaptive Measur	and Fast Net ements	work-wide
Measure **sate Ke <b>ABSTENCE</b> Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained Constrained	Elastic : Elastic : Tech yangtoo Dimitic Tech mianti ABSTRACT	Sketch: Adaptive Measur	and Fast Net ements resolution resolution ne Sketch to letwork Flow	x a la la com work-wide
Measur * state to * state to ESTEME ESTEME Hereouth and here angle the angle more angle the an	Elastic : Elastic : Final Control Cont	utomated Statisti Sketch: Adaptive Measur ausure O Rethinking M Zaosing Lar, Antonio	and Fast Net ements ne Sketch to letwork Flow Manousis-, Gregory V	work-wide
Measur * state is * state is <b>USTEM</b> How the state state is a state state state is a state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state state states state states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states states	Elastic : Elastic : Tree yangton Quantum Tech Martin Tech Martin Tech Martin Tech Martin Tech Martin Tech Martin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hartin Hart	Sketch: Adaptive Measur acts to Rethinking M Zaoxing Liu, Antone	and Fast Net ements ne Sketch to letwork Flow Manousis-, Gregory V	e work-wide ward lage 1 of 10 Rule Them All: Monitoring with U organger, Vyas Sakar, Vadimi Carnegie Mellon University
Measure *Bate Ro *Bate Ro *Bate Ro Berner Howk and how any the how any the	Elastic : Elastic : Field yundton Q Nuttur Tech Main Ni ABSTRACT Won stered, by indicate any dist the field of the stered the ste	Sketch: Adaptive Measur Control of Control Rethinking N Zaoning Lui, Antonia Joint ABSTRACC	and Fast Net ements ements emerster ne Sketch to letwork Flow Manousis-, Gragory V Manousis-, Gragory V	e work-wide mean and the first of the firs
Measure *late k *late k HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HST	Elastic : Elastic : Frederic Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter Verter	Sketch: Adaptive Measur Control of Control Rethinking N Zaoning Lui, Antonia Joint ABSTRACC	and Fast Net ements ements emerster ne Sketch to letwork Flow Manousis-, Gragory V Manousis-, Gragory V	
Measure *late k *late k HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HSTRM HST	Elastic : Elastic : Final Control Co	Sketch: Adaptive Measur Constraints Measur Constraints Constraints Measur Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Constraints Cons	and Fast Net ements • spend of the letter of the second of the second of the letter of the second of the letter of the second of the Manouslis - Gregory V Manouslis - Gregory V Manouslis - Gregory V Manouslis - Gregory V	e work-wide wear lange 1 of 1 Rule Them All: Monitoring with U oranger, Vyas Sakar, Vadim 'Carnegie Mellon University' 1 Introduction Narden Inshifteden The Saka Sakar, Madrid Carnegie Mellon University'
Measure *area to the second	Elastic S Elastic S Elasti	Sketch: Adaptive Measur Statisti Sketch: Adaptive Measur Statisti Construct Austract Newsmagneticity and Abstract Newsmagneticity and Abstract	and Fast Net ements representation ne Sketch to letwork Flow Manousis-, Gregory V mas Hopkins University accurate estimates of met- fing and representing ca- ter in the second second second accurate estimates of met- fing and representing ca- ter in the second second second second accurate estimates of met- fing and representing ca- ter in the second second second second accurate estimates of met- man definition of met- metric accurate second second second second accurate estimates of met- metric accurate second second second second accurate estimates of met- metric accurate second second second second accurate estimates of metric accurates of metric accurate estimates of metric accurates of metric accurates of metric accurates of metric a	Control of the second sec
Measure *area to the second	Elastic : Elastic : Fed yeldo Control of the second Control of	Sketch: Adaptive Measur Control of the second Rethinking M Zaoring Lu', Antonio M Zaoring Lu', Antonio M Zaoring Lu', Antonio M Zaoring Lu', Antonio M Zaoring Lu', Antonio M Zaoring Lu', Antonio M M M M M M M M M M M M M M M M M M M	and Fast Net ements : generate at the Sketch to letwork Flow Manousie: Grogory V Manousie: Grogory V Manou	
Measure *base *base Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner Horner	Elastic : Elastic : Feed yangton Quantum Tech Mananu Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Commun	Sketch: Adaptive Measur Stats of Rethinking N Zaosing Lut, Antonis 1 Joint Zaosing Lut, Antonis 1 Joint Market Imagement replicit Stats of AltsTRACT	and Fast Net ements • expension • expensio	
Measure * the second s	Elastic : Elastic : Figure : Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support Support	Sketch: Adaptive Measur Sketch: Adaptive Measur atsia Control Rethinking M Zaosing Lu, Anton Jacobia ABSTRACT News magnetic nepton for the server spectra	and Fast Net ements	Contract of the second seco
Measure * the second s	Elastic : Elastic : Trend Particle Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Const	Sketch: Adaptive Measur Sketch: Adaptive Measur Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant Constant	and Fast Net ements ements : execute the second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second second sec	A second
Measure * the second s	ement with A	utomated Statisti → extense Sketch: Adaptive Measur action Control (1) Control (1) Contr	and Fast Net ements ements encode and the second emergence of the second emergence of the second emergence of the second emergence of the second manual second the second emergence of the	Control of the marked of
Measure * the second s	Elastic : Elastic : Feed yangton Quantum Tech Mananu Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Communication Commun	Sketch: Adaptive Measure Sketch: Adaptive Measure 313 5 20 Rethinking M Zaoing Lui, Anon 'Joi NETRACT New Anony decision Anony decision for the second provide the second of the second provide the second of the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provi	and Fast Net ements and Fast Net ements • vouwers of the Sketch to letwork Flow Manousis- Grogory V Manousis- Grogory V ms Hopfins University accurate cellulato of est- fing cap, engineering of est- engineering of est- engineering of est- engineering of est- engineering of est- engineering of est- engineering of est- fing cap, engineering of est- engineering of est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est- est-	A stable start and start a
Measure * the second s	ement with A	Sketch: Adaptive Measure Sketch: Adaptive Measure To a second second Rethinking M Zaosing Liv, Antonis Cassing Liv	and Fast Net ements and Fast Net ements • spendorski ne Sketch to lettwork Flow Manousis', Grogory V Manousis', Grogory V Ma	Contract of the second se
Measure * the second s	ement with A	Sketch: Adaptive Measure Sketch: Adaptive Measure 313 5 20 Rethinking M Zaoing Lui, Anon 'Joi NETRACT New Anony decision Anony decision for the second provide the second of the second provide the second of the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provide the second provi	and Fast Net ements ements : execute the second of the second execute the second of the second of the second technologies of the second of the	Control of the second sec
Measure * the second s	ement with A	Sketch: Adaptive Measure Sketch: Adaptive Measure ats to CREMING M Zanxing Luk, Atoms And State And State	and Fast Net ements and Fast Net ements • execution •	Control of the second sec
Measure * the second s	ement with A	statutionated Statisti () () () () () () () () () () () () () (	and Fast Net ements and Fast Net ements • execution •	
Measure * the second s	ement with A	steetch: Adaptive Measure Sketch: Adaptive Measure Is I I Comparison Rethinking M Zaoing Lu', Angel Zaoing Lu', Angel M Zaoing Lu', Angel Ju Anstract M Anstract Sketch is Adaptive for print state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of the state of t	and Fast Net ements and Fast and a	Contract of the second seco
Measure * the second s	ement with A	steerer of the second s	and Fast Net ements and Fast Net ements represent represent merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merit	A A A A A A A A A A A A A A A A A A A
Measure *arate k *arate k BETHE BETHE How has not any any any any term of the set term of term of	ement with A	steerer of the second s	and Fast Net ements and Fast Net ements represent represent merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merits merit	A and a second a s
Years to year to start to star	ement with A	utomated Statisti ↓ entropy Sketch: Adaptive <u>Measur</u> 303 10 <u>Measur</u> 200 10 <u>Measur</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u> <u>Aussi</u>	and Fast Net ements and Fast Net ements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rement	Contract of the second se
Years to year to start to star	ement with A	steerer of the second s	and Fast Net ements and Fast Net ements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rements rement	A A A A A A A A A A A A A A A A A A A

## Current monitoring methods are inadequate

- Not fast enough
  - Involve CPU and control planes
  - Network state changes rapidly

## Do not provide end-to-end state

 Difficult to correlate per-element state with the actual path of a flow

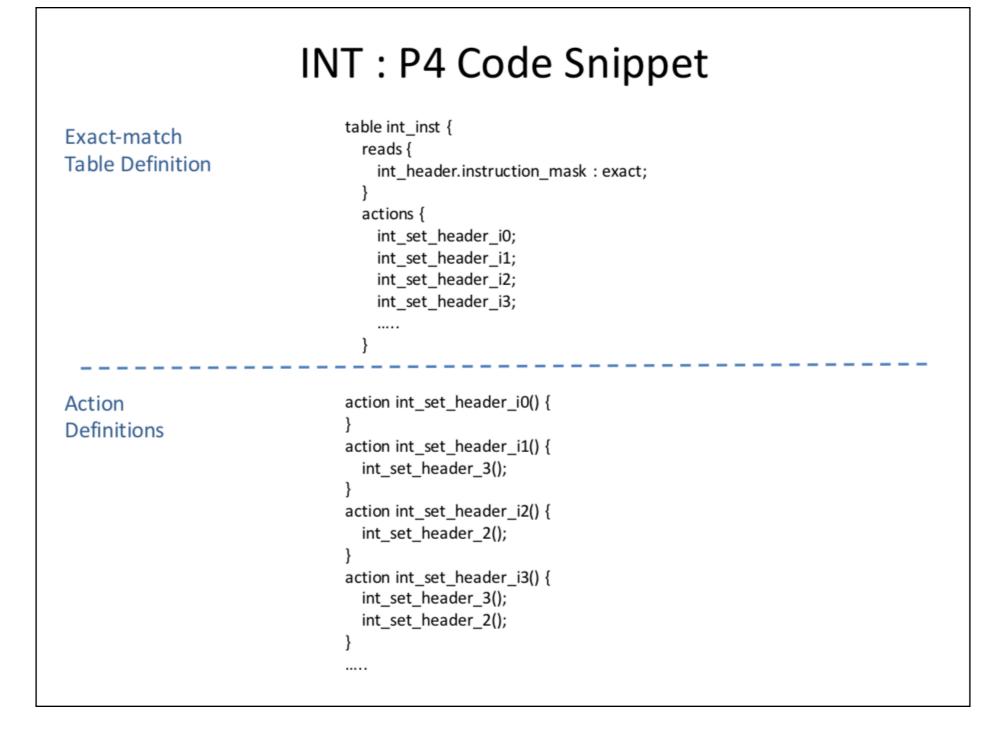
## INT : In-band Network Telemetry

- Mechanism for collecting network state in the dataplane
  - As close to realtime as possible
  - At current and future line rates
  - With a framework that can adapt over time
- Examples of network state
  - Switch ID, Ingress Port ID, Egress Port ID
  - Egress Link Utilization
  - Hop Latency
  - Egress Queue Occupancy
  - Egress Queue Congestion Status
  - .....

			4 B	ytes		
Ve	r	Flags	Instruction Count	Max Hop Count	Total Hop Count	Metadata
Ins	struc	tion Bitma	ip	Reserved		Header
0			Most Recent I	NT Metadata		
0			INT Me	tadata		
						_Metadata
1			First INT N	Metadata		

# INT using P4

- P4 enables flexible packet parsing and modification for INT
- P4 allows INT to adapt to
  - Any Encapsulation format
  - Any State required to be collected
  - Any feature, protocol current and future



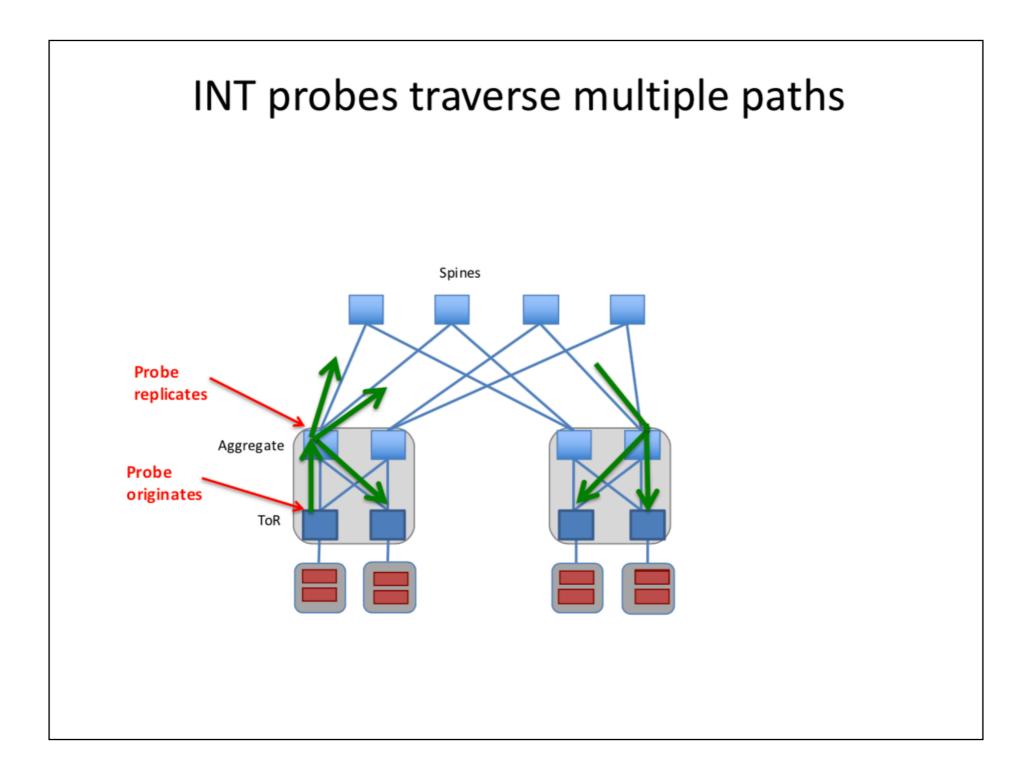
## HULA: INT + Flowlet routing

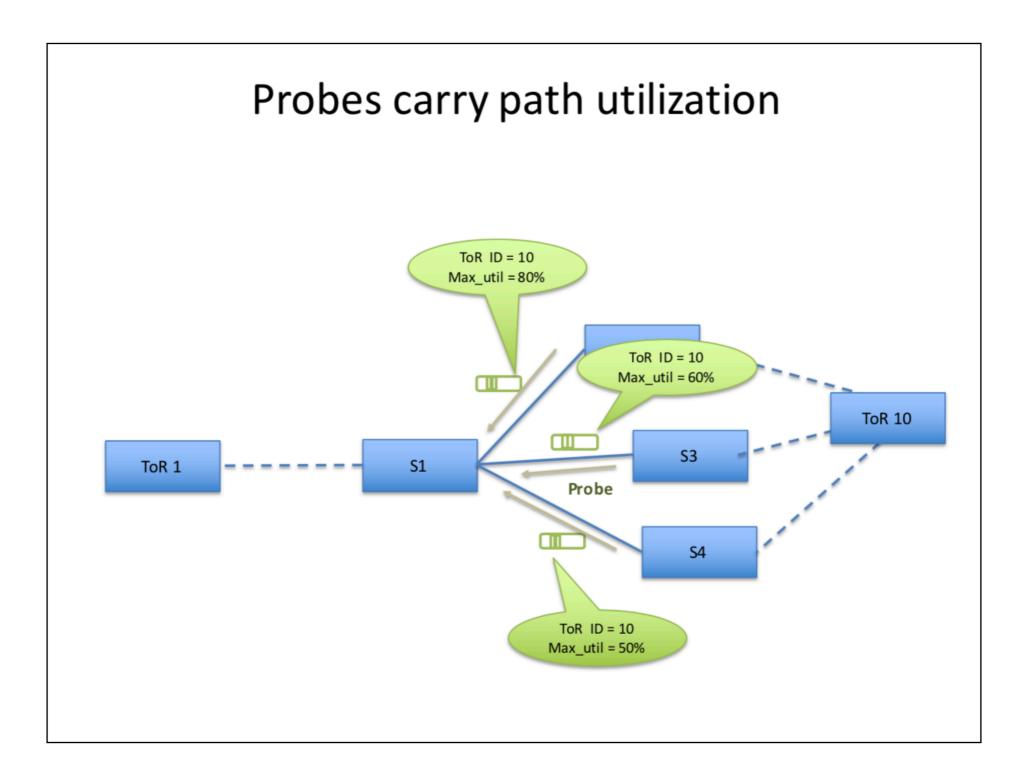
## 1. Periodic INT probes

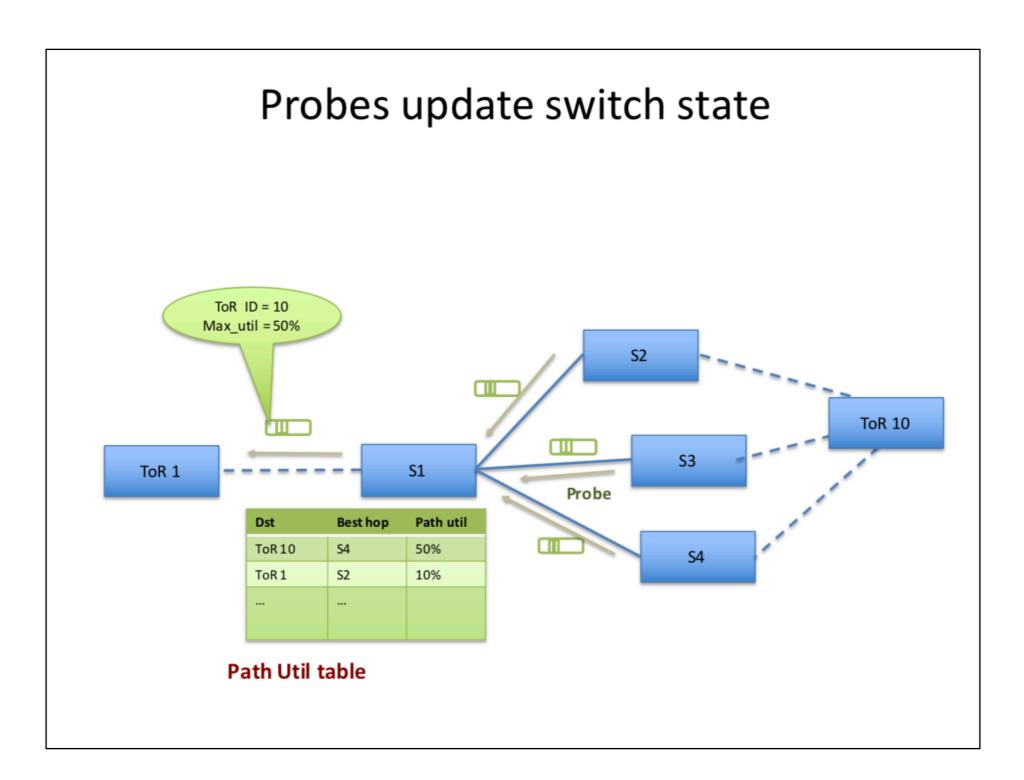
disseminate path utilization to switches

# 2. Flowlet detection and path selection

- happens at all switches
- hop-by-hop adaptive routing









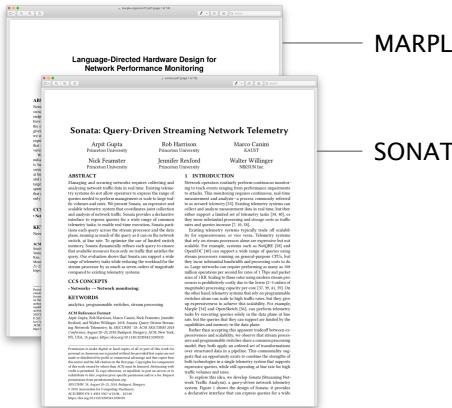
- INT provides real-time network state directly in the dataplane
  - Scales to arbitrarily large networks
  - Scales to current and future link speeds
  - Can adapt to any network, any encap, any application
- Knowledge of real-time network state opens up new possibilities
  - Enhanced monitoring and troubleshooting
  - Network-state aware routing
  - •

	Language-Directed I Network Perform	ance Monitoring	r
	a a b	🚊 sonata.pdf (page 1 of 15)	🗶 * 👌 💿 Q. Starth
AB5 Netw			
swite			
endp			
the c	Country One Dat		A
gives we as	Sonata: Query-Dri	iven streaming N	etwork relemetry
expre	Arpit Gupta	Rob Harrison	Marco Canini
that i varie	Princeton University	Princeton University	KAUST
w			
milia is ba	Nick Feamster Princeton University	Jennifer Rexford Princeton University	Walter Willinger NIKSUN Inc.
swite at lin	,	,	
and s	ABSTRACT	1 INTROD	
targe	Managing and securing networks require analyzing network traffic data in real time.		ors routinely perform continuous monitor- ts ranging from performance impairments
that s	try systems do not allow operators to exp	ress the range of to attacks. This :	monitoring requires continuous, real-time
only	queries needed to perform management or s		ad analysis—a process commonly referred metry [55]. Existing telemetry systems can
CC	fic volumes and rates. We present Sonata, a scalable telemetry system that coordinates		ze measurement data in real time, but they
• Net	and analysis of network traffic. Sonata provi	ides a declarative either support a	limited set of telemetry tasks [34, 40], or
	interface to express queries for a wide ra telemetry tasks, to enable real-time executi		intial processing and storage costs as traffic s increase [7, 10, 58].
KE	tions each query across the stream proces	sor and the data Existing telen	netry systems typically trade off scalabil-
Netw	plane, running as much of the query as it ca switch, at line rate. To optimize the use o		reness, or vice versa. Telemetry systems am processors alone are expressive but not
ACM Srinis	memory, Sonata dynamically refines each		imple, systems such as NetQRE [58] and
Venka	that available resources focus only on traffic		an support a wide range of queries using
Kim. Monit	query. Our evaluation shows that Sonata ca range of telemetry tasks while reducing the	n support a wide stream processo workload for the they incur substa	rs running on general-purpose CPUs, but initial bandwidth and processing costs to do
27-25 https:/	stream processor by as much as seven orde	ers of magnitude so. Large networ	ks can require performing as many as 100
and a construction	compared to existing telemetry systems.		is per second for rates of 1 Tbps and packet ling to these rates using modern stream pro-
	CCS CONCEPTS	cessors is prohibi	tively costly due to the lower (2-3 orders of
Permi	<ul> <li>Networks → Network monitoring;</li> </ul>		essing capacity per core [37, 39, 41, 59]. On elemetry systems that rely on programmable
for pro	KEYWORDS	switches alone c	an scale to high traffic rates, but they give
author republi and/or	analytics, programmable switches, stream		ss to achieve this scalability. For example, OpenSketch [56], can perform telemetry
SIGCE	ACM Reference Format:	tasks by executi	ng queries solely in the data plane at line
© 201' for Co ACM	Arpit Gupta, Rob Harrison, Marco Canini, Nick Resford, and Walter Willinger. 2018. Sonata: Qua	Feamster, Jennifer rate, but the quer	ies that they can support are limited by the
ACM1 https:/	ing Network Telemetry. In SIGCOMM '18: AC	M SIGCOMM 2018 Dath and I	nemory in the data plane. ccepting this apparent tradeoff between ex-
	Conference, August 20–25, 2018, Budapest, Hungar NY, USA, 15 pages. https://doi.org/10.1145/3230	y. nc.m, New York, 543 3210555 pressiveness and	scalability, we observe that stream proces-
		sors and program	mable switches share a common processing h apply an ordered set of transformations
	Permission to make digital or hard copies of all or personal or classroom use is granted without fee provid		h apply an ordered set of transformations data in a pipeline. This commonality sug-
	made or distributed for profit or commercial advantage	and that copies hear gests that an opp	ortunity exists to combine the strengths of
	this notice and the full citation on the first page. Copyr of this work owned by others than ACM must be honor	red. Abstracting with expressive querie	s in a single telemetry system that supports es, while still operating at line rate for high
	credit is permitted. To copy otherwise, or republish, to	post on servers or to traffic volumes a	nd rates.
	redistribute to lists, requires prior specific permission permissions from permissions@acm.org.	To explore un	s idea, we develop Sonata (Streaming Net-
	SIGCOMM '18, August 29–25, 2018, Budapest, Hangary 9: 2018. Association for Computing Machinery.		alysis), a query-driven network telemetry shows the design of Sonata: it provides
	ACM ISBN 978-1-4503-5567-4/18/08\$15.00		erface that can express queries for a wide

Loss	Stadar: Fast Detection of Lost P	Z • 2 © Q term
	Center Networks	liang.pdf (page 2 of 16)
		🗶 💌 👌 🛞 Q. Sauch
ABSTR. Packet loss caused by a and have si network of detection o We also nee information these losse are generic fall short in	Yuliang Li* Rui Miao* C	etFlow for Data Centers hangboon Kim <sup>*</sup> Minlan Yu <sup>*</sup>
low overhe: we propose detection. V individual I entire netw	*University of Southern Calij	fornia <sup>1</sup> Barefoot Networks
on prototyp is easy to :	Abstract	plement NetFlow in hardware is how to maintain an ac-
memory an information	NetFlow has been a widely used monitoring tool with	tive working set of flows using a data structure with low time and space complexity. We need to handle collisions
analysis to: with a few (	a variety of applications. NetFlow maintains an active working set of flows in a hash table that supports flow insertion, collision resolution, and flow removing. This is hard to implement in merchant silicon at data cen-	during flow insertion and remove old flows to make room for new ones. These tasks are challenging given the lim- ited per-packet processing time at merchant silicon. To handle this challenge, today's NetFlow is imple-
1. INT	ter switches, which has limited per-packet processing time. Therefore, many NetFlow implementations and	mented in two ways: (1) Using complex custom silicon
Packet loss	other monitoring solutions have to sample or select a	that is only available at high-end routers, which is too
happen for keynote [7]	subset of packets to monitor. In this paper, we observe	expensive for data centers; (2) Using software to count sampled packets from hardware, which takes too much
for one yea	the need to monitor all the flows without sampling in	CPU resources at switches. Because of the lack of us-
losses, 4 n	short time scales. Thus, we design FlowRadar, a new way to maintain flows and their counters that scales to a	able NetFlow in data centers, operators have to mirror
random cor immediate	large number of flows with small memory and bandwidth	packets based on sampling or matching rules and ana-
significant	overhead. The key idea of FlowRadar is to encode per-	lyze these packets in a remote collector [26, 40, 44, 34]. It is impossible to mirror all the packets because it takes
especially v	flow counters with a small memory and constant inser- tion time at switches, and then to leverage the computing	too much bandwidth to mirror the traffic, and too many
	tion time at switches, and then to leverage the computing power at the remote collector to perform network-wide	storage and computing resources at the remote collector
Permission to a or classroom a	decoding and analysis of the flow counters. Our eval-	to analyze every packet. (Section 2)
distributed for and the full cit	uation shows that the memory usage of FlowRadar is	However, in data centers, there is an increasing need to have visibility of the counters for all the flows all the
owned by othe mitted. To cop	close to traditional NetFlow with perfect hashing. With	to have visibility of the counters for all the nows all the time. We need to cover all the flows to capture those tran-
lists, requires g	FlowRadar, operators can get better views into their net- works as demonstrated by two new monitoring applica-	sient loops, blackholes, and switch faults that only hap-
coNEXT 'I	tions we build on top of FlowRadar.	pen to a few flows in the Network and to perform fine-
© 2016 ACM.	1 Introduction	grained traffic analysis (e.g., anomaly detection). We need to cover these flows all the time to identify transient losses, bursts, and attacks in a timely fashion. (Section 3)
		In this paper, we propose FlowRadar, which keeps
	NetFlow [4] is a widely used monitoring tool for over 20 years, which records the flows (e.g., source IP, destina-	counters for all the flows with low memory overhead and exports the flow counters in short time scales (e.g.,
	years, which records the nows (e.g., source iP, destina- tion IP, source port, destination port, and protocol) and	and exports the now counters in short time scales (e.g., 10 ms). The key design of FlowRadar is to identify the
	their properties (e.g., packet counters, and the flow start-	best division of labor between cheap switches with lim-
	ing and finish times). When a flow finishes after the in-	ited per-packet processing time and the remote collector
	active timeout, NetFlow exports the corresponding flow	with plenty of computing resources. We introduce en-
	records to a remote collector. NetFlow has been used for a variety of monitoring applications such as accounting	coded flowsets that only require simple constant-time in-
	a variety of monitoring applications such as accounting network usage, capacity planning, troubleshooting, and	structions for each packet and thus are easy to implement with merchant silicon at cheap switches. We then decode
	attack detection.	these flowsets and perform network-wide analysis across
	Despite its wide applications, the key problem to im-	time and switches all at the remote collector. We make

	B INT-current-spec.pdf (page 1 of 28) INT-current-spec.pdf (page 1 of 28) INT-current-spec.pdf (page 1 of 28)
	In-band Network Telemetry (INT)
	Changhoon Kim, Parag Bhide, Ed Doe: <i>Barefoot Networks</i> Hugh Holbrook: <i>Arista</i> Anoop Ghanwani: <i>Dell</i> Dan Daly: <i>Intel</i> Mukesh Hira, Bruce Davie: <i>VMware</i>
l	
Process I Header	Ing INT Headers IT Header Types Iandling INT Packets Format and Location IT over any encapsulation

er: Data Plane	Performance Diagn		
Q Q 1	😒 sosrt8-final3	6.pdf (page 1 of 7)	🗶 🔹 👌 🛞 Q Search
Net	work-Wide Heavy		tion with
		ty Switches	
	tob Harrison, Qizhe Cai, Arp Princeton	vit Gupta, and Jennife University	er Rexford
ABSTRACT Many network mon that stand out a g	itoring tasks identify subsets of traffic	60 -	5m Interval 60s Interval
Protocol Independent identify these "heavy	op-k flows for a particular statistic. A t Switch Architecture (PISA) switch can hitter" flows directly in the data plane,	(j) 40 -	10s Interval
ing against a thresho want to identify inte	c statistics across packets and compar- old. However, network operators often resting traffic on a <i>network-wide</i> basis.	<sup>∞</sup> 20 -	he he see a
wide visibility, we pr tion scheme for netw	ween line-rate monitoring and network- resent a distributed heavy-hitter detec- orks modeled as one-big switch. We use	2 10	20 30 40 50 ampling Rate (1/x)
ing directly in the da	to perform efficient threshold monitor- ta plane. We implement our system us- and evaluate it using real-world packet ate that our solution can accurately de-	Figure 1: This graph shows between two major ISPs [12] w	i the recall for detecting heavy-hitters ith different monitoring intervals. Even call quickly diminishes and worsens as
tect network-wide h communication over	eavy hitters with up to 70% savings in head compared to an existing approach	the monitoring interval decrea	ases.
with a provable uppe	er bound.	ory and processing overh accuracy, in order to deal w	data structures, that bound mem- ead in exchange for some loss in with the limited resources available
Network operators of traffic, to detect atta	ften need to identify outliers in network :ks or diagnose performance problems.	on the switches. While prior work has f	ocused on heavy-hitter detection
hitter" detection that creding a pre-determ	tect unusual traffic is to perform "heavy identifies the top-k flows (or flows ex- ined threshold), according to some met-	and superspreaders [27] o	k operators often need to track the s. For example, port scanners [15] ould go undetected if the traffic is ation. Detecting the beavy hitters
ric. For example, net receiving traffic from high-precision in ord	work operators often track destinations a large number of distinct sources with ler to detect and mitigate DDoS attacks	not sufficient. Large flows	ation. Detecting the heavy hitters and then combining the results is s can easily fall "under the radar" till heave sinchle tests solume. Are
or TCP incast [4] in heavy-hitter detection	real time. In traditional networks, this on relies on analyzing packet samples Programmable switches open up new	plying a lower detection the chance of missing larg	still have sizable total volume. Ap- threshold at each switch reduces ge flows, at the expense of higher
possibilities for aggro large flows directly i	egating traffic statistics and identifying n the data plane [17, 18, 24, 27]. These	communication overhead dinator. Additionally, networks	to report counts to a central coor- that forward high traffic volumes
Permission to make digit personal or classroom use made or distributed for pri this notice and the full circ	al or hard copies of all or part of this work for is granted without fee provided that copies are not dif or commercial advantage and that copies hear tion on the first page. Copyrights for components ers than ACM must be honored. Abstracting with y otherwise, or emploish, to poot on on servers or to es prior opencific permission and/or a fee. Request jons/hear next.	often resort to sampling defined based on the needs	1/x packets, where x is operator- s of the specific network. However.
of this work owned by oth credit is permitted. To cop redistribute to lists, requir	est than ACM must be honored. Abstracting with y otherwise, or republish, to post on servers or to es prior specific permission and/or a fee. Request	Figure 1, we show the impa	abstantially reduced accuracy on a when traffic volumes are high. In act sampling has on accuracy while detection on a link between two
permissions from permiss SOSR '18, March 28–29, 20 © 2018 Association for Ce ACM ISBN 978-1-4503-560	18 Los Anodos CA USA	major ISPs [12] processing Even with high sampling	g approximately 1 GBps of traffic. rates, recall is quite low on short
ACM ISBN 978-1-4503-560 https://doi.org/10.1145/31	54-0/18/05\$15.00 85467.3185476	decrease. In modern data	kly diminishes as sampling rates center networks where switches
	<ul> <li>p576-huang.pdf (page 1 of 15)</li> </ul>		
		🗶 = 👌 🛞 Q Search	
rement with /	ng User Burdens in Automated Statist	ical Inference	
	Automated Statist	n Approximat ical Inference	
rement with A	Automated Statist	n Approxima ical Inference raystatis	Z = B O Q herd
Elastic	Automated Statist	n Approxima ical Inference reserves and Fast Net ements	v v v v v v v v v v v v v v v v v v v
Elastic	Automated Statist	n Approxima ical Inference reserves and Fast Net ements	e V • D © Q terro work-wide
Elastic	Automated Statist	n Approxima ical Inference reserves and Fast Net ements	v v v v v v v v v v v v v v v v v v v
Elastic	Sketch: Adaptive Measur	n Approxima ical Inference resorters and Fast Net ements	vork-wide
Elastic Flastic	Sketch: Adaptive Measur	n Approxima ical Inference report of the and Fast Net ements - superstant ne Sketch to	v v v v v v v v v v v v v v v v v v v
Elastic Elastic Unitive Tech yungdor O Institue Tech haar Haar A Minoru Abstract	Sketch: Adaptive Measur	n Approxima ical Inference and Fast Net ements ements enerts	e work-wide measure tage rates Rule Them All: Monitoring with U
Elastic Elastic Tek yangton Q hattu Tec haan Minora A Minora ABSTRACT When network is an attach, Dov	Sketch: Adaptive Measur	n Approxima ical Inference and Fast Net ements _ spentian ne Sketch to letwork Flow	work-wide
Elastic Elastic	Sketch: Adaptive Measur	n Approxima ical Inference and Fast Net ements _ spentian ne Sketch to letwork Flow	work-wide
Elastic Elastic United States States States Elastic States States States Elastic States States Elastic States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States States Stat	Sketch: Adaptive Measur Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Contro	n Approximaa ical Inference filestation and Fast Net ements second fast second fast ne Sketch to Network Flow	e work-wide work-wide weight large 1418 weight
Elastic Elastic Tech yangtor Q Initia Tech Main Tech Main Tech Mai	Sketch: Adaptive Measur Rethinking M Zaoxing Liu, Antonie J ABSTRACT Nerverk mangement reptres republication, and accept (eg.	n Approximat ical Inference Prove 14 10 and Fast Net ements 	e work-wide work-wide weathing of the second weathing of the second weath
Elastic Elastic Tech yangon Q Initiu Tech Man Han Man Han Han Han Han Han Han Han Han Han H	Automated Statist	n Approximat ical Inference Prove 1475 and Fast Net ements : superstant ne Sketch to letwork Flow Manousis, Gregory V Manousis, Gregory V accessed and convergence Manousis, Strey of superstant accessed and anony of superstant accessed and anony of superstant accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed accessed acces	
ement with <i>J</i>	Automated Statist	n Approximat ical Inference Prove 14 10 and Fast Net ements 	Contract of the second se
Elastic CSCRCCE CSCRCE	Automated Statist	n Approximati ical Inference Provide 1998 and Fast Net ements • second state tements • second state net Sketch to ketwork Flow • Manousis · Gregory V • Manousis · Gregory · Manousis · Gregory V • Manousis · Gregory · Manousis · Gr	Control of the second sec
Elastic Elastic Technological Second Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control C	Automated Statist	n Approximation ical Inference and Fast Net ements • second a second a second • second a second a second a second • second a second a second a second • second a second a second • seco	Conception of the second
Elastic Elastic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classic Classi	Automated Statist	n Approximation ical Inference Fuertor	Control of the second sec
ement with <i>J</i> The second sec	Automated Statist	an Approximation ical Inference Fault of the second second second second and Fast Net ements net Sketch to lettwork Flow Manual State Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second Second S	Contract of the second se
ement with <i>J</i>	Automated Statist	n Approximation ical Inference and Fast Net ements • second a second second a second term of the second a second second a second a second term of the second a second second a second a second term of the second a second second a second a second term of the second a second second a second a second term of the second a second term term of the second a second a second term of the second a second a second a second term of the second a second a second a second term of the second a second a second a second a second term of the second a second a second a second a second term of the second a second a second a second a second a second term of the second a second a second a second a second a second term of the second a secon	Contract and the formation of the second secon
Elastic Elastic Elastic Telestic Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter Peter	Automated Statist	n Approximation ical Inference and Fast Net ements • second a second second a second term of the second and second a second a second term of the second a second second a second a second term of the second a second term of the second a second second a second a second term of the second a second a second a second term of the second a second a second a second term of the second a second a second a second a second term of the second a second a second a second a second term of the second a second a second a second a second term of the second a second a second a second a second term of the second a second a second a second a second a second a second term of the second a	Contract of the second se
Elastic Elastic Functional Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction	Automated Statist	n Approximat ical Inference Turner of the and Fast Net enents and Fast and F	
Elastic Elastic Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Co	Automated Statist Automated Statist Sketch: Adaptive Measur Control Measur Measur Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Cont	n Approximation ical Inference runt and Fast Net emensor emersor sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector	Conception of the section of th
Elastic Elastic Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Construction Co	Automated Statist Automated Statist Sketch: Adaptive Measur Control Measur Measur Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Cont	n Approximation ical Inference runt and Fast Net emensor emersor sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector	
The second secon	Automated Statist Automated Statist Sketch: Adaptive Measur Control Measur Measur Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Control Cont	n Approximation ical Inference runt and Fast Net emensor emersor sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector sector	Contract of the second se



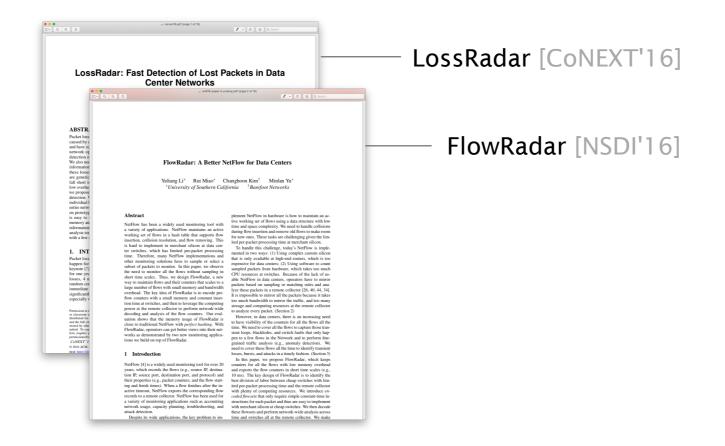
MARPLE [SIGCOMM'17]

SONATA [SIGCOMM'18]

Both papers enable operators to express monitoring queries

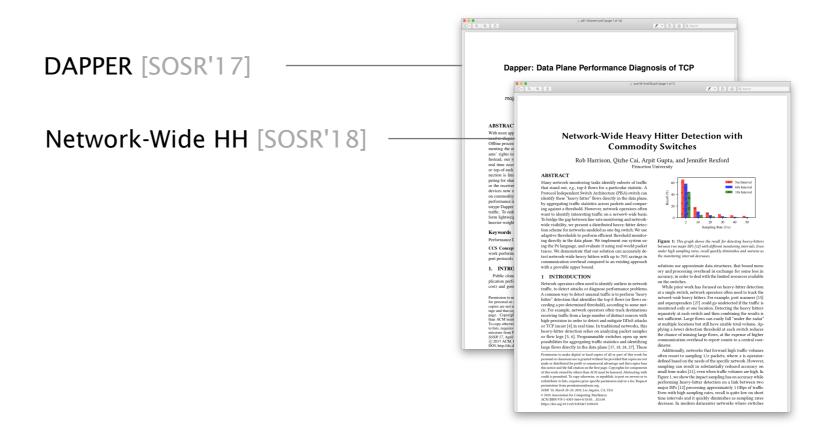
A compiler then compiles these queries to: switch programs + control code

The two papers differ among others in the types of queries they support



Develop techniques and tools to monitor *all flows* by

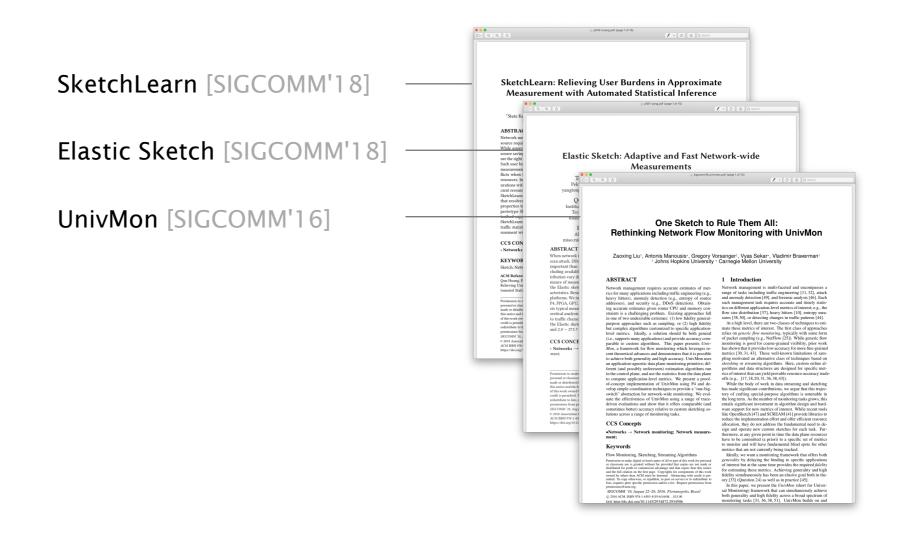
- relying on in-switch data structures (Bloom Filters) and
- decoding them at the controller-level



Develop P4-based detection mechanisms to

- diagnose TCP performance issue (e.g. small receiver buffers)
- heavy-hitter (e.g. port scanners, superspreader, DDoS)

Introduce techniques to make sketch-based monitoring more practical (by making sketches adaptive or "universal")



Data plane for programmability

Performance Monitoring Applications offloading

Platforms Correctness Management for Data plane programmability

#### [SOSR'15]

	.sosr15.pdf (page 1 of 7)		17.pdf (page 1 of 7)		sigcomm17-silkroad.pdf (page 1)	
۵.	Z · 👌 🛞 Q Search	0. Q. Q. A	X × D @ Q Search			
NetDevees Concern	and at Natural Canad		ation is a Dumb Idea			
NetPaxos: Consen	sus at Network Speed		e Has Come			Load Balancing Fast
			bdelaziz, Abdulla Aldilaijan,		Cheap Using Switch	ing ASICs
Huynh Tu Dang	g* Daniele Sciascia*		i, Panos Kalnis	D : 10		
Marco Canini <sup>†</sup> Ferna	ndo Pedone" Robert Soulé"	KA	UST	Rui Miao University of Southern Cal	ifornia Facebook	Changhoon Kir Barefoot Network
<sup>*</sup> Università della Svizzera italiar	na <sup>†</sup> Université catholique de Louvain	ABSTRACT	Programmable networks create the opportunity for in-	University of Southern Car	iorina racebook	Bareloot Network
		Programmable data plane hardware creates new opportuni-	network computation, i.e., offloading a set of compute opera-		Jeongkeun Lee	Minlan Yu
		ties for infusing intelligence into the network. This raises a	tions from end hosts into network devices such as switches	I	Barefoot Networks	Yale University
		fundamental question: what kinds of computation should be	and smart NICs. In-network computation can offer substan-	ABSTRACT	ACM F	Reference format:
BSTRACT	grammability", allowing ordinary programs to manage the	delegated to the network?	tial performance benefits, as it is for example the case with consensus protocols [9, 10] and in-network caches [20]. Al-	In this paper, we show that up		ao, Hongyi Zeng, Changhoon Kim, Jeongkeun
s paper explores the possibility of implementing the wide	network through direct access to network devices.	In this paper, we discuss the opportunities and challenges	though traditional networks are not capable of computation,	In this paper, we show that up load balancer (SLB) servers can	n he replaced by a single lan Yu.	2017. SilkRoad: Making Stateful Layer-4 I
loyed Paxos consensus protocol in network devices. W	e Several recent projects have used SDN platforms to demon-	for co-designing data center distributed systems with their network layer. We believe that the time has finally come for	the idea of using the network s are not explaine of computation,	modern switching ASIC, potential	lly reducing the cost of load COMM	nd Cheap Using Switching ASICs. In Proce 1 '17, Los Angeles, CA, USA, August 21–25,
ent two different approaches: (i) a detailed design de	strate that applications can benefit from improved network	offloading part of their computation to execute in-network.	to perform computation on transmitted data is reminiscent	balancing by over two orders of m	agnitude. Today, large data https://	/doi.org/10.1145/3098822.3098824
ption for implementing the full Paxos logic in SDN swite		However, in-network computation tasks must be judiciously	of Active Networks [30], which proposed to replace packets	centers typically employ hundreds load-balance incoming traffic over		
ch identifies a sufficient set of required OpenFlow exten		crafted to match the limitations of the network machine archi-	with small programs called "capsules" that are executed at	software load balancers (SLBs)		
is; and (ii) an alternative, optimistic protocol which ca	n and the state of	tecture of programmable devices. With the help of our exper-	each traversed switch. However, for the past two decades	service (with a virtual IP address,		NTRODUCTION
implemented without changes to the OpenFlow API, but ies on assumptions about how the network orders mes		iments on machine learning and graph analytics workloads,	the hardware capabilities were lacking. This appears to be	tasked with providing the service		ul layer-4 (L4) load balancers scale out s
es on assumptions about now the network orders mes es. Although neither of these protocols can be fully im	Non-Advantage for the second for the test of the second seco	we identify that aggregation functions raise opportunities to	changing. The recently proposed RMT architecture [6] and its upcom-	addresses, or DIPs). An SLB is st		d datacenters by mapping packets destin
mented without changes to the underlying switch firmwa		exploit the limited computation power of networking hard-	ing incarnation in the Barefoot Networks' Tofino [3] switch	a connection to the same server, changes and/or if the load is sp		virtual IP address (VIP) to a pool of
argue that such changes are feasible in existing hardware	words: how can distributed applications and protocols uti-	ware to lessen network congestion and improve the overall application performance. Moreover, as a proof-of-concept,	chip has a flexible parser and a customizable match-action	pool. This property is called per		ble direct IP addresses (DIPs or DIP p ing is a critical function for inbound traff
reover, we present an evaluation that suggests that mov	lize network programmability to improve performance?	we propose DAIET, a system that performs in-network data	engine. To process packets at high speed, this architecture has	PCC. The challenge is that the los	ad balancer must keep track and tra	affic across tenants. A previous study [36
Paxos logic into the network would yield significant per	This paper focuses specifically on the Paxos consensus	aggregation. Experimental results with an initial prototype	a multi-stage pipeline where packets flow at line rate. Each	of millions of connections simulta	aneously. an aver	erage of 44% of cloud traffic is VIP tra
mance benefits for distributed applications.	protocol [19]. Paxos is an attractive use-case for several reasons. First, it is one of the most widely deployed pro-	show a large data reduction ratio (86.9%-89.3%) and a similar	stage has a fixed amount of time to process every packet,	Until recently, it was not pos balancer with PCC in a merchan	in the Actual Ac	load balancing function. Building cloud-
	tocols in highly-available, distributed systems, and is a fun-	decrease in the workers' computation time.	allowing for lookups in memory (SRAM and TCAM), manip-	high-performance switching ASICs	Datatici	ing faces two major challenges: rt full bisection traffic with low latency:
tegories and Subject Descriptors	damental building block to a number of distributed applica-		ulating packet metadata and stateful registers, and performing boolean and arithmetic operations using ALUs. Other ven-	per-connection states with PCC		rt full bisection traffic with low latency: apid growth in traffic; doubling every yea
2.4 [Distributed Systems]: Network operating systems			dors are also introducing new classes of programmable chips	provide resources and primitives	to enable PCC at a large [11] an	nd growing by 50 times in six years in
[Performance of Systems]: Reliability, availability, and	d on optimizing Paxos [20, 22, 31, 32], which suggests that	1 INTRODUCTION	with similar capabilities [7]. We believe that with this new	scale. In this paper, we explore how		the community has made efforts to sca
viceability; D.4.5 [Reliability]: Fault-tolerance	the protocol could benefit from increased network support.	The advent of flexible networking hardware [6] and expres-	generation of flexible data plane hardware it is worth revis-	build much faster load balancers t Our system, called SilkRoad, is	Virtual	l switching to match full bisection bandw
	Third, moving consensus logic into network devices would	sive data plane programming languages [5, 29] have produced	iting a fundamental question: as networks become capable	program and when compiled to a	uatacci	nter traffic (or full gateway capacity for ir ), one missing piece is scaling L4 load balant
ywords	require extending the OpenFlow API with functionality that	networks that are deeply programmable. The functionality	of computation, what kinds of computation should networks	ASIC, we show it can load-balan		l bisection bandwidth of the underlying ph
tware-defined networking, Paxos, NetPaxos.	is amenable to an efficient hardware implementation [3, 5]. Implementing Paxos in the network provides a different	of networks can now be enriched without hardware changes	perform?	simultaneously at line rate.		balancing is also a critical segment for the
<u>,</u>	point in the design space, and identifies a different set of net-	while retaining the capability of processing packets at very	In this paper, we will consider this question in the scope	CCS CONCEPTS		mance of delay-sensitive applications [2]
INTRODUCTION	work requirements for protocol implementors. This paper	high rates, even above Terabits per second. Emerging pro-	of data center applications because it is likely that data cen- ters will be early adopters of programmable networks and			y data centers (e.g., $2-5 \mu s RTT$ with R
oftware-defined networking (SDN) is transforming th		grammable network devices are paving the way for new ser-	many of these applications have stringent performance re-	<ul> <li>Networks → Programmable networks;</li> </ul>		e per connection consistency (PCC) du pol changes: Data center networks are con
retworks are configured and run. In contrast to tradi		vices to better support data center applications [9, 18] and improve network monitoring [13, 16, 24–26].	quirements. On the one hand, in-network computations can	ment, Data tenter networks,		handle failures, deploy new services, upp
al networks, in which forwarding devices have propri	ment the full Paxos logic in SDN switches; and (ii) an alter-	improve network monitoring [15, 10, 24–20].	be broadly useful in several performance-oriented contexts to	KEYWORDS		es, and react to the traffic increase [24
y control interfaces, SDNs generalize network device	native, optimistic protocol which can be implemented with-	<sup>‡</sup> Amedeo Sapio is also with Politecnico di Torino.	reduce latency and/or increase throughput of certain opera-	Load balancing; Programmable s		l change can result in many DIP pool
g a set of protocols defined by open standards, including		Permission to make digital or hard copies of all or part of this work for	tions. Furthermore, it can help reducing network traffic, so as		exampi	le, when we upgrade a service, we need t and upgrade them one by one to avoid aff
st prominently the OpenFlow [24] protocol. This mov		personal or classroom use is granted without fee provided that copies are not	to alleviate congestion, which is a major cause of application	Permission to make digital or hard co		apacity. Such frequent DIP pool updates
ards standardization has led to increased "network pro-	Although neither of these protocols can be fully imple- mented without changes to the underlying switch firmware,	made or distributed for profit or commercial advantage and that copies bear	performance degradation. In particular, a computation that happens on-path and at line rate is appealing since it bears no	for personal or classroom use is gran	ted without fee provided that from a	a large web service provider with about
sission to make digital or hard copies of all or part of this work for personal	or we appear avidence to show that such aban are are fassible	this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstractine	happens on-path and at line rate is appealing since it bears no cost to the application, which can spare CPU cycles for other	copies are not made or distributed for and that copies bear this notice and	the full citation on the first data ce	enter clusters (§3.1).
sroom use is granted without fee provided that copies are not made or distribute profit or commercial advantage and that copies bear this notice and the full citation	a contraction of the second	with credit is permitted. To copy otherwise, or republish, to post on servers or	tasks instead. On the other hand, despite recent technological	page. Copyrights for components of the ACM must be honored. Abstracting wi	his work owned by others than Duri	ing a DIP pool change, it is critical t ction consistency (PCC), which means a
he first page. Copyrights for components of this work owned by others than th	ing consensus logic into the network would reduce appli-	to redistribute to lists, requires prior specific permission and/or a fee. Request	advancements, network devices have limited compute power	otherwise, or republish, to post on ser	rvers or to redistribute to lists.	ction consistency (PCC), which means a numerical should be delivered to the same
or(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or blish, to post on servers or to redistribute to lists, requires prior specific permission	or cation complexity, reduce application message latency, and	permissions from permissions@acm.org. HotNets-XVI, November 30–December 1, 2017, Pulo Alto, CA, USA	and little storage to support general computation. Moreover,	requires prior specific permission and from permissions@acm.org.	packets	is of an ongoing connection to a different
for a fee. Request permissions from Permissions@acm.org.	increase transaction throughput.	© 2017 Copyright held by the owner/author(s). Publication rights licensed to	systems designers are prescribed by the end-to-end princi-	SIGCOMM '17, August 21-25, 2017,	Los Angeles, CA, USA the con	onnection. It often takes subseconds to
R2015, June 17 - 18, 2015, Santa Clara, CA, USA syright is held by the owner/author(s). Publication rights licensed to ACM.	In summary, this paper makes the following contributions:	Association for Computing Machinery.	ple [28] to avoid implementing application-specific logic in	© 2017 Association for Computing Ma ACM ISBN 978-1-4503-4653-5/17/08.	. \$15.00 applica	ations to recover from a broken connecti
tyright is held by the owner/author(s). Publication rights licensed to ACM. M 978-1-4503-3451-8/15/06\$15.00.		ACM ISBN 978-1-4503-5569-8/17/11\$15.00 https://doi.org/10.1145/3152434.3152461		https://doi.org/10.1145/3098822.3098	.24 second	l in Wget), which significantly affects use
//dx.doi.org/10.1145/2774993.2774999.						

Consensus at network speed

#### In-Network Aggregation

Stateful layer-4 load balancers

(e.g., for MapReduce, graph analytics, ML)

+ NetCache [SOSP'17], NetChain [NSDI'18]

### [HotNets'17]

#### [SIGCOMM'17]

Data plane for programmability

Performance Monitoring Applications offloading

Platforms Correctness

for Data plane programmability

Management

"Data-plane" programmability goes beyond switch programmability (or P4 for that matter)

## Offloading...

## ... to FPGA-based SmartNICS

#### host networking

age, and petabytes of formance, both ban cloud workloads, (

	# (mar 0 of #1)
<ul> <li>nsdit8-firestone.pr</li> </ul>	df (page 2 of 15)
Azure Accelerated Networking:	
Mike Andrewartha Hari Angepat Vivel Harish Kumar Chandrappa Somesh Chaturmohta Fengfen Liu Kalin Ovtcharov Jitu Padhye G Mark Shaw Gabriel Silva Madhan Sivakumar Ni	Gautham Popuri Shachar Raindel Tejas Sapre sheeth Srivastava Anshuman Verma Qasim Zuhair Vaid David A. Maltz Albert Greenberg
Abstract	all virtual networking features, such as private virtual net- works with customer supplied address spaces, scalable L4
Modem cloud architectures rely on each server running its was networking stack to implement policies such as tun- leiling for virtual networks, security, and load balancing. However, these networking stacks are becoming increasing the complex as features are added and as network speedby processing power from VMs, increasing the coard run- ting cloud services, and adding latency and variability to stroyd performance. We present Azure Accelerated Networking (AccelNet), ary solution for offloading host networking to hard variability to stroyd performance. We present Azure Accelerated Networking to AccelNet, ary solution for offloading host networking to hard vari- ning cloud services, and adding our networking to the service strong the service of the service of the service of the service service services with the service of the service of the service services with the service of the service of the service of the services with the service service service and the service services with the service service service services and the service of the service servers since late 2015 in a level < 15 JM Nott PCI heaterisc and 2020 his number level of services the fastest network avail- able to costoners in the public cloud. We present the lexing in AccelNet, including our hardware co- losing model, performance results on key workloads, and de-speriores and lessons learned from developing and de- hoying AccelNet and PGA bases.	lead balancers, security groups and access control lists (ACLs), virtual tradies, bandwicht metering, QoS, and mer. These features are the responsibility of the host platform, which typically means software running in the topervisor. The cost of providing these services continues to in- rease, in the span of only a few years, we increased ne- vel address of the services of the services of the output of the services of the services of the output of the services of the services of the top services of the services of the services of the posterior services. The services of the services of the top services additional CPU cycles. Burning CPUs for these services that seaws from the processing capabilities, running this stack in software or ab- providing cload services. Single Root I/O Virtualization (SR-IOV) [4,5] has been reposed to reduce CPU ultication by allowing direct as- cess to NIC hardware from the VM. However, this di- ter access would byposs the host SDN stack, making the MC responsible for implementing all SDN splices, services that services the content of the services reasonable to the services to the services. In this paper we present Azure Accelerated Herwork- ing (AccelNet), our host SDN stack, implemented on the FDA based Area SmarkNIC. AccelNet provides near- native network performance in a virtualized environment, foldading packet processing in the host CPU to the Azure SmarkNIC. Building upon the software h-sead VIP in botts SDN platicines [6], and the hardware and software and - software model in the software h-sead VIP in the ASURE SMARCH and the software h-sead VIP in the ASURE SMARCH and the software h-sead VIP in the ASURE SMARCH and the software h-sead VIP in the ASURE SMARCH and the software h-sead VIP in the
I Introduction	frastructure of the Catapult program [7, 8], AccelNet pro- vides the performance of dedicated hardware, with the
The public cloud is the backbone behind a massive and apidly growing percentage of online software services [], [, 3]. In the Microsoft Azure cloud alone, these services onsume millions of processor cores, exabytes of stor- age, and petabytes of network bandwidth. Network per-	programmability of software nunning in the hypervisor. Our goal is to present both our design and our experiences running AccelNet in production at scale, and lessons we learned.
ormance, both bandwidth and latency, is critical to most	2 Background
cloud workloads, especially interactive customer-facing workloads. As a large public cloud provider, Azure has built its	2.1 Traditional Host Network Processing In the traditional device sharing model of a virtualized environment such as the public cloud, all network I/O to and form a churing during in an elevine and environment in the

#### congestion control

#### HotCocoa: Hardware Congestion **Control Abstractions**

Monia Ghobadi Jennifer Rexford Microsoft Research Princeton University

Tahmasbi Arashloo Princeton University ABSTRACT

Mina

••• •• • •

ontrol in multi-tenant data centers is an active ch because of its significant impact on customer and, consequently, on revenue. Therefore, new d protocols are expected to emerge as the Cloud lowing new consection control alevorithms in the equent updates, but pro vervisor and implementing rol algorithm, such as traf-oftware have well-studied ies. In this paper, we argue ontrol algorithm in ble NICs. To do so, we identify the absence of a simple high-level dona. as a broad set of congestion sedware implementation. It about low-level hardware primit ives. To HotCocca, we implement four congestion control ns (Reno, DCTCP, PCC, and TIMELY) and use sim-o show that HotCocca's implementation of Reno perf Reno per

s the behavior of a native imple

#### 1 INTRODUCTION

congestion control (CC) algorithms play a central role ta center network's efficiency and its tenants' quality erience. Hence, a significant number of congestion l algorithms concentrate on data center networks, which customizing their infrastructure to serve oads and tenants [3, 4, 8, 9, 23, 29, 31, 33], to continue, given the impact of network e and their rapid adoption

puting Machinery. 9-8/17/11...\$15.00

to the NIC (e.g., TCP Si and Generic Rec [2]). More recent te uch as Single Root I/0 NAT\_ACLs\_etc.) to the NICs [15]. We t

even then, it is challenging and time hm. Thus, it takes

nd deploy a new CC als ingestion contact, and algorithms to enable higher-level abstractions that give operations and the second s structure across different CC definition of higher-level abstra enable th

## NetFPGA SUME board

[NSDI'18]

and from a physical d

[HotNets'17]



# David Walker tors may deploy their CC algorithms in t While this approach enables frequen implementation, it incurs well-studied doing correction

# Host-based programmability + SmartNICs + programmable switches = fully programmable platforms

Big question is

## How to combine them best?

#### ••• ••• • • • beyond\_smart\_nics.pdf (page 1 of 6) 🗶 👻 📩 🛞 Q Search

#### **Beyond SmartNICs:** Towards a Fully Programmable Cloud

Adrian Caulfield Microsoft Research acaulfie@microsoft.com

(Invited Paper) Paolo Costa Microsoft Research pcosta@microsoft.com

Monia Ghobadi Microsoft Research mgh@microsoft.com

application to hardware and networks. In this paper, we focus on the potential of FPGA-based SmartNICs and programmable switches to realize this vision and illustrate some of the research challenges that need to be addressed to fully unleash its benefits.

I INTRODUCTION

However, the compute cycles-measured by the number of switches, e.g., [15], [16], this opens up exciting opportunities CPU cycles required to process each packet-are falling to rethink the way in which we design and deploy applications behind the massive acceleration in available network band-and network functions. We argue that we should move away width [3]. As a result, CPU time is increasingly becoming a contributor to the per-packet latency in high-speed cloud data application functions towards a fully programmable cloud, centers. To make matters more challenging, modern clouds in which application logic can be distributed across multiple are embracing a fully software-defined network (SDN) and accelerators and network devices. are increasingly expected to perform complex network policies such as regular expression matching and encryption [4]. Such applications that run in it. These benefits include application policies drive up the per-packet CPU cycles, which in turn specific control of network flows, the ability to run code increases the cloud costs and adds unpredictable latency to at precisely the right location in the network hierarchy, and the cloud services.

Traditionally, this has been addressed by offloading various a large-scale machine learning workload, a neural network networking functions to the Network Interface Cards (NICs) running on a distributed set of SmartNICs would benefit from such as TCP Segmentation Offload [5] and Generic Receive the direct interface to the network to reduce inference and Offload [6]. However, these techniques are not flexible enough training latencies. Further, programmable switches running to support complex policies. Techniques such as SR-IOV en- custom flow management code can reduce latency and optiable VMs to bypass the hypervisor and send packets directly to mize bandwidth by scheduling flows in an application-specific the NIC [7]. PCIe Process Address Space ID (PASID) reduces way, improving efficiency. Finally, the switch could even host the hardware resource requirements of SR-IOV, enabling it a parameter server, directly performing aggregation of the to scale to support containers or even individual processes. training weights from the SmartNICs below it. But, these techniques bypass the hypervisor, making it hard to After summarizing the key technology underpinning Smartenforce SDN-like policies.

as a new platform that enables network operators a flexible however, requires solving a number of novel and exciting environment to offload complex network policies and maintain research questions, which we outline in Section IV.

Abstract--FPGA-based SmartNICs and programmable switches have been recently introduced to leverage hardware acceleration and custom pipelines inside the cloud infrastructure. These devices are capable of handling the per-packet processing needs at line rate, including load balancing, encapsulation, congestion management, and security. We argue, however, that the benefits provided by these new devices could extend beyond software-defined networking use cases and they prompt a shift hardware-software co-design across all layers, ranging from application to hardware and networks. In this paper, we focus platforms and programming languages around FPGA-based SmartNICs [4], [12]-[14].

Most prior work treats the FPGA and the NIC domains separately by either focusing on the FPGA capabilities as a generic device or focusing on the NIC functions. In this paper, we turn our attention into the combined domains and argue that an FPGA-based SmartNIC should be thought of as both The continuous growth of cloud applications [1] is driving a programmable FPGA-based accelerator and a networking a steady increase in network infrastructure's bandwidth [2]. device. When combined with recently proposed programmable

A fully programmable cloud provides significant benefits to direct, low latency access to the network. For example, in

NICs in Section II, we describe our vision underlying the fully Recently, FPGA-based SmartNICs have been introduced programmable cloud in Section III. Implementing this vision,

IEEE International Conference on High Performance Switching and Routing, 2018 Data plane for programmability

Performance Monitoring Applications offloading

Platforms Correctness for Data plane programmability

Management

## So you've a programmable networks...

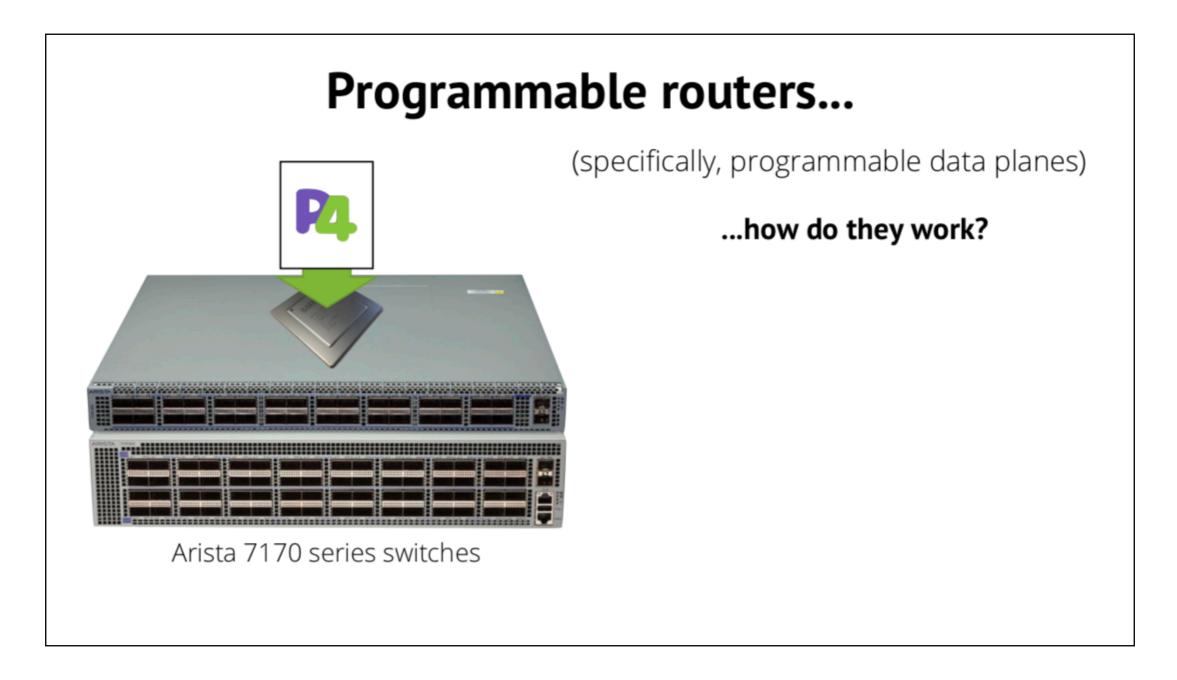
## How do you make sure that it works as it should?!

		Debugging P4 programs with Vera	[SIGCOMI
p4v: Practical Verification fo	r Programmable Data Planes*	Radu Stoenescu Dragos Dumitrescu Matei Popovici Lorina Negrean Costin Raiciu University Politehnica of Bucharest firstname.lastname@es.pub.ro	iu I
Barefoot Networks Yale University Barefoot Ithaca, NY, USA New Haven, CT, USA Santa Cla Robert Soulé Han Wang Călin ( University of Lugano Barefoot Networks Barefoot	Networks a. CA, USAMilad Sharif Bardool Networks Stand Clar. CA, USAJongkeun Leg Bardool Networks Stand Clar. CA, USAActional CA, USAStandorl CA, USANate Foster Correl University Itaka, NY, USAActional CA, USAStandorl CA, USANate Foster 	<section-header><section-header><section-header><text><text><text><text><text><text><text><text><text><text><text><text><text><text><text><text><text><text></text></text></text></text></text></text></text></text></text></text></text></text></text></text></text></text></text></text></section-header></section-header></section-header>	ritine net- Defa and he effects momentum halts and has should dia corre- hymat aff (super left 18) rams in Feasible Time sertions Schaeffer-Filho, Marinho Barcellos mormatics

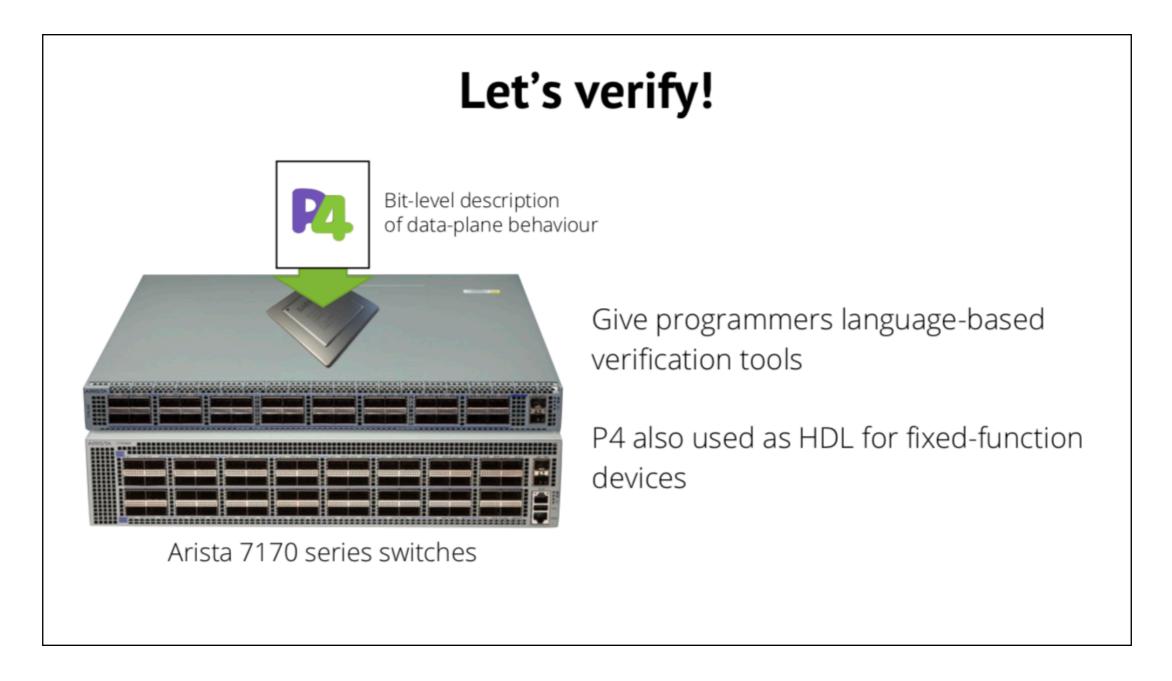
## So you've a programmable networks...

## How do you make sure that it works as it should?!

			g P4 programs with Vera	[SIGCOI
p4v: Practical Verification for	Programmable Data Planes*		Costin Raiciu iversity Politehnica of Bucharest firstname.lastname@cs.pub.ro	
Jed Liu William Hallahan Cole Sch Barefoot Networks Yale University Barefoot Y Ithaca, NY, USA New Haven, CT, USA Santa Clara	letworks Barefoot Networks Barefoot Networks	bolic execution. Vera automatically uncover common bugs including parsing/deparsing/		ie net- ta and effects entum
Robert Soulé Han Wang Călin C. University of Lugano Barefoot Networks Barefoot N Lugano, Switzerland Santa Clara, CA, USA Santa Clara	letworks Stanford University Cornell University	Vera can also be used to verify user-specified j novel language we call NetCTL. To enable scalable, exhaustive verification o crossobate Vera sourcementscalls userances all us	properties in a behind programmable networks, we expect such faul many others will cripple programmable networks.	
ABSTRACT	1 INTRODUCTION	cessing optimized for verification. These tech	iniques allow programs both before deployment and at runtime. At it	
We present the design and implementation of p4v, a prac- tical tool for verifying data planes described using the P4	Suppose you wanted to verify the correctness of a network data plane. How would you do it? One approach, which	the execution of a purely symbolic packet in. program currently available (6kLOC)		(range 1 of 13)
programming language. The design of p4v is based on classic verification techniques but adds several key innovations including a novel mechanism for incorporating assumptions about the control plane and domain-specific optimizations which are needed to scale to large programs. We present case studies showing that p4v verifies important properties and finds bugs in real-world programs. We conduct experiments to quantify the scalability of p4v on a wide range of additional examples. We show that with just a few hundred lines of control-plane annotations, p4v is able to verify critical safety properties for switch. p4, a program that implements the functionality of on a modern data center switch, in under three minutes. <b>CCS CONCEPTS</b> . Networks $\rightarrow$ Programming interfaces; • Software and its engineering $\rightarrow$ Software verification: <b>KEYWORDS</b> Programmable data planes, P4, verification. ACM Reference Format: Jed Liu, William Hallahan, Cole Schlesinger, Milad Sharif, Jeongkeun Lee, Robert Soulé, Han Wang, Cálin Caşcaval, Nick McKeown, and Nate Foster. 2018, P4V: Practical Verification for Programmable	is widely used today, is to rely on exhaustive testing—i.e., generate a set of input packets and test whether the device produces the expected outputs. Testing is expensive, since modern devices handle dozens of different packet formats and protocols, each requiring distinct test inputs. But with a conventional device these costs are paid only once, because its capabilities are "baked in" at manufacturing time and cannot be changed by programmers. Recently, the field has started to shift to more flexible plat- forms in which data-plane functionality is not controlled by vendors but can be defined by programmers. The idea is that the programmer describes the functionality of the device using a program in a domain-specific language such as P4 [5, 44, 45], and the compiler generates an efficient im- plementation for the underlying target device. This approach not only facilitates rapid innovation, since new protocols can be deployed without having to spin new hardware, it also opens up opportunities for novel uses of the network—e.g., in- band network telemetry [26] and in-network caching [28, 29] to name a few. While increased programmability offers ben- efits, it also creates challenges related to correctness. <i>Example.</i> Consider a "bump in the wire" firewall that uses acl and nat tables to filter and rewrite incoming packets	phe concrete dataplanes at once by all to insert symbolic table entries, the highlights possible control plane error We have used from https://post. bgs in each of them in secondermin CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC	Recent trands in software-defined networking have extended net- work programmability to the data plane. Unfortunative, the chance of introducing bags increases significantly. Verification can help preven bags by saming that the program does not visible its re- cepters properties and to spalidy verify complex invations. In this a none general Poverficiation supports. Developers, annotate PA programmer to result in straffords the verification agreements a none general Poverficiation supports. Developers, annotate PA programmer to result is straffordered into C models and all possi- ble paths symbolically executed. We implement a prototype, and use it to show the foundition of the verification agreement. Because symbolic carecturing one of the verification agreement is provided by the properties and to all constraints, the properties of the We use the possible care exceeded to show the gains provided by an experiment provide the soft static result is standification where speed us bechniques (save of constraints, program slicin, pro- platilication), and experiment with different complexities Possible and out in the sham an innive considering variants Pagnification.	Intions  http://www.intionality.com/output/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/second/s
Data Planes. In SIGCOMM '18: SIGCOMM 2018, August 20–25, 2018, Budapest, Hungary. ACM, New York, NY, USA, 14 pages. https: //doi.org/10.1145/3230543.3230582	to verify that if acl is populated with rules that drop packets going to a given internal host, the host will be isolated from the external network. Even for this simple property, several complications can arise, illustrating the need for verification. First, the behavior of the program that implements the firewall may be undefined on certain kinds of packets since, according to the P4 language specification [44], reading or		NULL WORKSO     New York Constraints, Programmable Data Planes     ACM Reference Format:     Mupul News, Loan Prinz, Alberto Schneffer Filhe, Marinho Barrellon,     201, Nerdication of P1 Programs in Feasible Tines using Americans, In     The With International Conference on merging Neurosite gal Deprinsion and     Technologies (CAMXT '10), December 4-2, 2014, Hendlam, Genera, ACM,     New York, NY, US, It pages Alberty-Robinsor g10.1105/2014(1.1238)421     INTRODUCTION	we show how a variety of speed up techniques can be employed to be added to the variation line and number of executed instructions. here techniques consist of using annotations in code to constrain the paths to be traversed according to properties and/or protocols interest, program slicing to reduce the complexity of the model and everification, and parallelization of symbolic execution. Be-
SIGCOMM '18, August 20–25, 2018, Budapest, Hungary © 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in SIGCOMM '18: SIGCOMM 2018, August 20–25, 2018, Budapest, Hungary, https://doi.org/10.1145/3230543.3220582.	writing an invalid header produces an arbitrary result. In par- ticular, although the acl table correctly matches and filters away IPv4 packets sent by external hosts, it might incorrectly forward other types of packets such as IPv6. Second, there is potential for confusion between internal and external ad- dresses. If the program executes the acl table before the nat table, then the rules intended to filter away external traffic		Data plane programmability allows operators to quickly deploy now a protocol and develop networks networks. Through programming, languages such as VA [2], ht possible to specify in a few instructions of the plane	dec, we experiment with order optimization features offered by urrent complex. To evaluate our approach, we built a prototype uning KLEE [4] on the P14Feiencer, complex [20] for the current language version, 4m. We applied it to four real P4 applications collected from the frances. Switch [21], MeVhand [3]. Despirit [11], and DCP [3]. Yar results show that the proposed verification process non nunever bend range of long celluler is the data place program intel <sup>4</sup> for it its method plane configuration. A detailed performance analysis also shows that, although the verification line grows exponentially with 2013 three methods the developed the sum to act the web.



Source: p4v, Practical Verification for Programmable Data Planes, Liu et al., 2018



### P4 by example

- P4 is a low-level language  $\rightarrow$  many gotchas
- Let's explore by example!
   IPv6 router w/ access control list (ACL)

```
control ingress { apply(acl); }
table acl {
  reads { ipv6.dstAddr: lpm; }
  actions { allow; deny; }
}
action allow() {
  modify_field(std_meta.egress_spec, 1);
}
action deny() { drop(); }
```

What could *possibly* go wrong?

### What if we didn't receive an IPv6 packet?

ipv6 header will be invalid

#### What goes wrong

Table reads arbitrary values → Intended ACL policy violated

Can read values from a previous packet  $\rightarrow$  Side channel vulnerability!

Real programs are complicated: hard to keep validity in your head

```
control ingress { apply(acl); }
```

```
table acl {
   reads { ipv6.dstAddr: lpm; }
   actions { allow; deny; }
}
```

```
action allow() {
   modify_field(std_meta.egress_spec, 1);
}
```

```
action deny() { drop(); }
```

### **Property #1: header validity**

### What if acl table misses (no rule matches)?

Forwarding decision is unspecified

### What goes wrong

Forwarding behaviour depends on hardware

- May not do what you expect!
- Code not portable

```
control ingress { apply(acl); }
table acl {
  reads { ipv6.dstAddr: lpm; }
  actions { allow; deny; }
}
action allow() {
  modify_field(std_meta.egress_spec, 1);
```

```
action deny() { drop(); }
```

### Property #2: unambiguous forwarding

### **Types of properties**

#### **General safety**

- Header validity
- Arithmetic-overflow checking
- Index bounds checking (header stacks, registers, meters, ...)

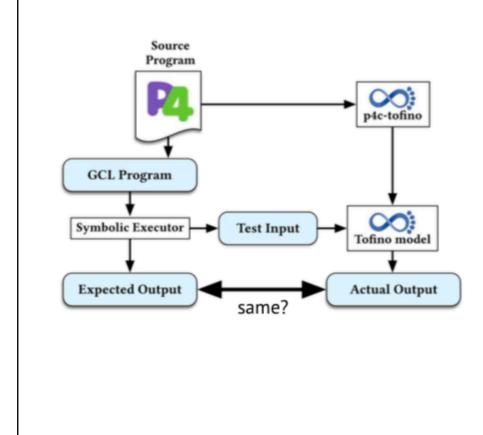
### Architectural

- Unambiguous forwarding
- Reparseability
- Mutual exclusion of headers
- Correct metadata usage (e.g., read-only metadata)

### Program-specific

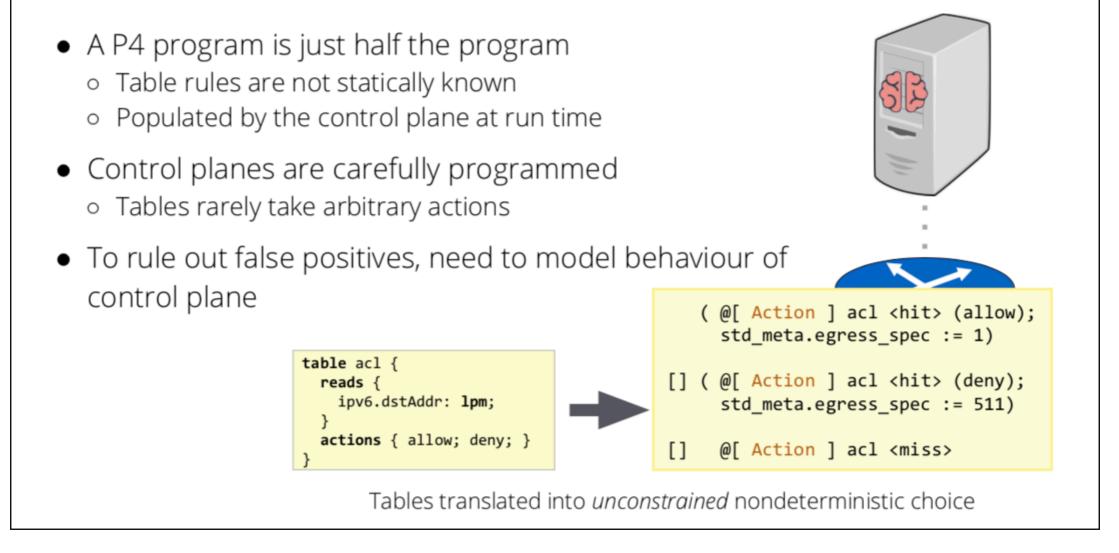
• Custom assertions in P4 program — e.g., IPv4 ttl correctly decremented

## Challenge #1: imprecise semantics

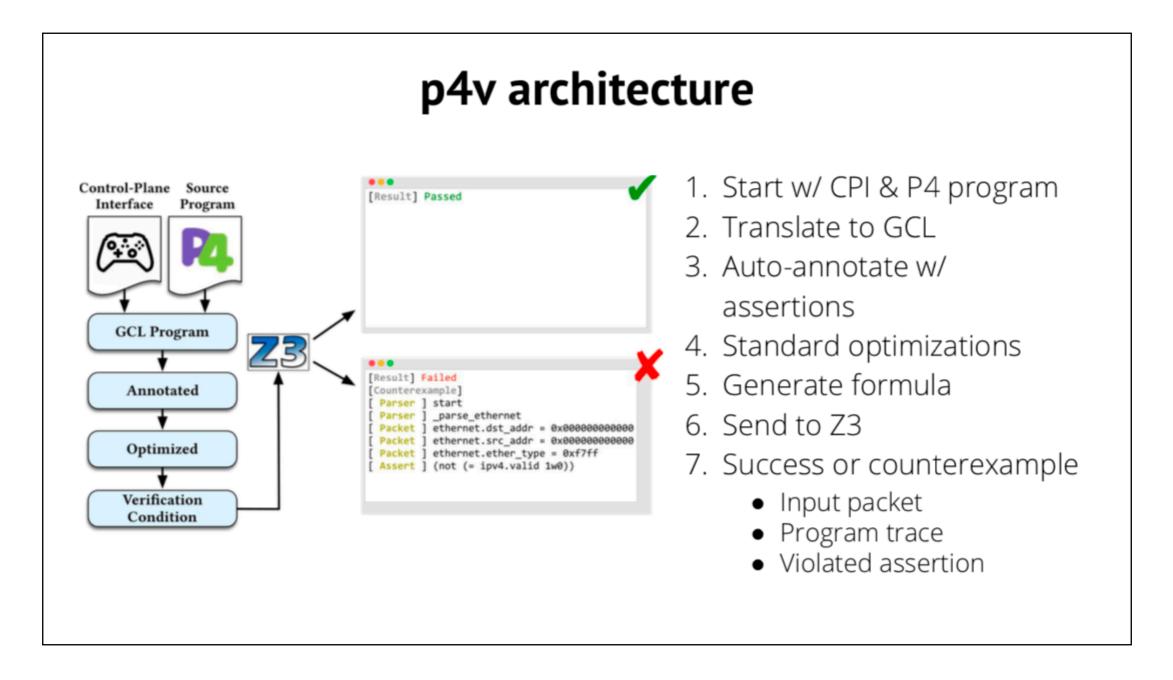


- P4 language spec doesn't give precise semantics
- Defined semantics by translation to GCL (a simple imperative language)
- Tested semantics
  - Symbolically executed GCL to generate input-output tests for several programs
  - Ran w/ Barefoot P4 compiler & Tofino simulator

## Challenge #2: modelling the control plane



## p4v overview • Automated tool for verifying P4 programs • Considers all paths • But also practical for **large programs** • Includes basic safety properties for any program • Extensible framework • Verify custom, program-specific properties • Assert-style debugging



Data plane for programmability

Performance Monitoring Applications offloading

Platforms Correctness

for Data plane programmability

Management

So you've a *verified* programmable networks... How do you manage it?!

How do you perform planned maintenance?

now that you've state in your switches...

How do you run multiple applications in your switches? monitoring, forwarding, load-balancing, etc.

How do you share resources amongst applications? especially memory and # packet operations

### We need an Operating System for the data plane

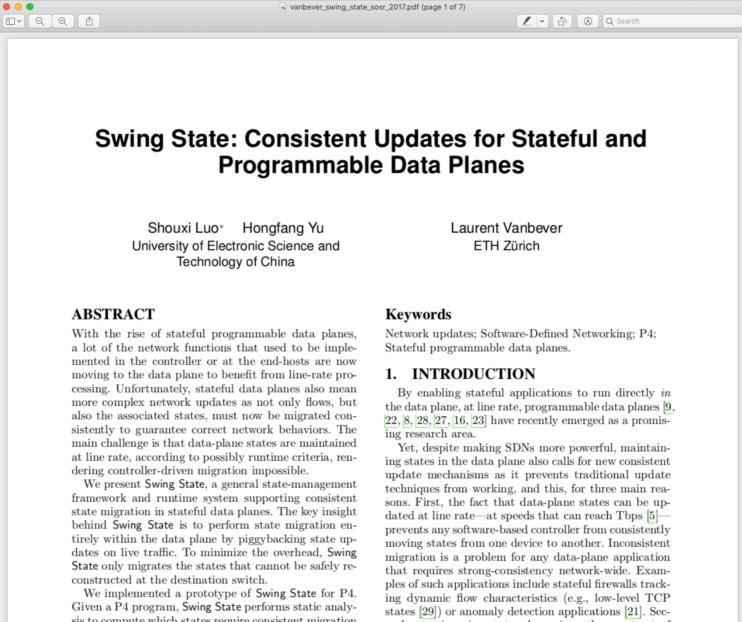
Definition Wikipedia

An operating system is a system software that manages computer hardware and software resources and provides common services for computer programs.

Do we have that? Nope. Not yet at least.

### We're working on it...

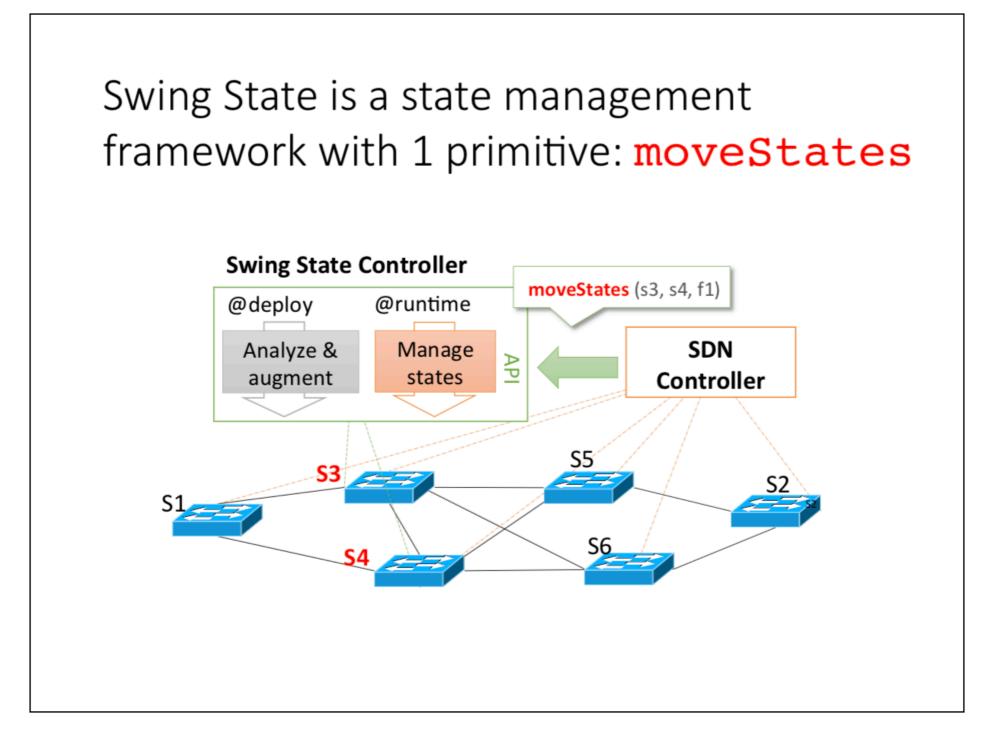
#### [SOSR'17]



sis to compute which states require consistent migration and automatically augments the program to enable the transfer of these states at runtime. Our preliminary results indicate that Swing State is practical in migrating data-plane states at line rate with small overhead.

#### **CCS** Concepts

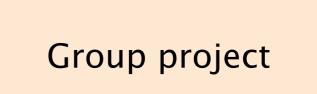
ond, even ignoring states dynamism, the exact set of states to be migrated may actually be unknown to the controller, preventing it from performing the migration in the first place. Indeed, the states location in memory can differ from device to device according to runtime factors (e.g. a hash computed on packet headers) that are invisible to the controller. Third, data-plane states



Source: Swing State: Consistent Updates for Stateful and Programmable Data Planes Luo et al., SOSR 2017

## Advanced Topics in Communication Networks





### ~7 weeks

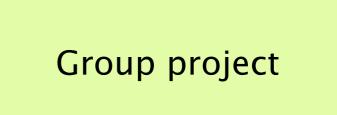
how to program in P4

>= 7 weeks

in teams of 2-3

## Advanced Topics in Communication Networks





### ~7 weeks

how to program in P4

>= 7 weeks

in teams of 2-3

## The group project starts this week It accounts for 50% of your final grade

implementation

70%

achieves the basic goals is properly documented runs...

implementationachieves the basic goals70%is properly documented<br/>runs...reportdescribes the main building blocks15%, 10 pages maxevaluates the solution<br/>describes what each group member did

implementation achieves the basic goals is properly documented runs... describes the main building blocks report evaluates the solution describes what each group member did summarizes the problem and the solution presentation contains a *live* demo 15%, 12 min. +questions involves all group members

## The final deadline for the project is Wed Dec 19 at 23.59pm

This week	Select a proposal from the list (see Doodle) or send us your own proposal by email
<i>Every</i> week	Meet with the responsible assistant schedule a recurring slot in [10.15am; noon]
Wed Dec 19 11.59pm	Send us an archive with report, code, slides
Thu Dec 20 8.15am—	Groups presentation + course/exam debrief attendance is mandatory

## The project has to be done in groups of 3 students "Matching" process for incomplete groups via Slack

Project grade is shared by each group member provided that each collaborated (roughly equally)

- Let us know in advance if that's *not* the case
- Briefly describe in the report the contribution of each group member
- Each group member should be involved in the presentation and be able to answer questions

### Details about each proposal is available on our website

#### Advanced Topics in Communication Networks Project Proposals

#### Proposal #1: Hardware-Based RSVP

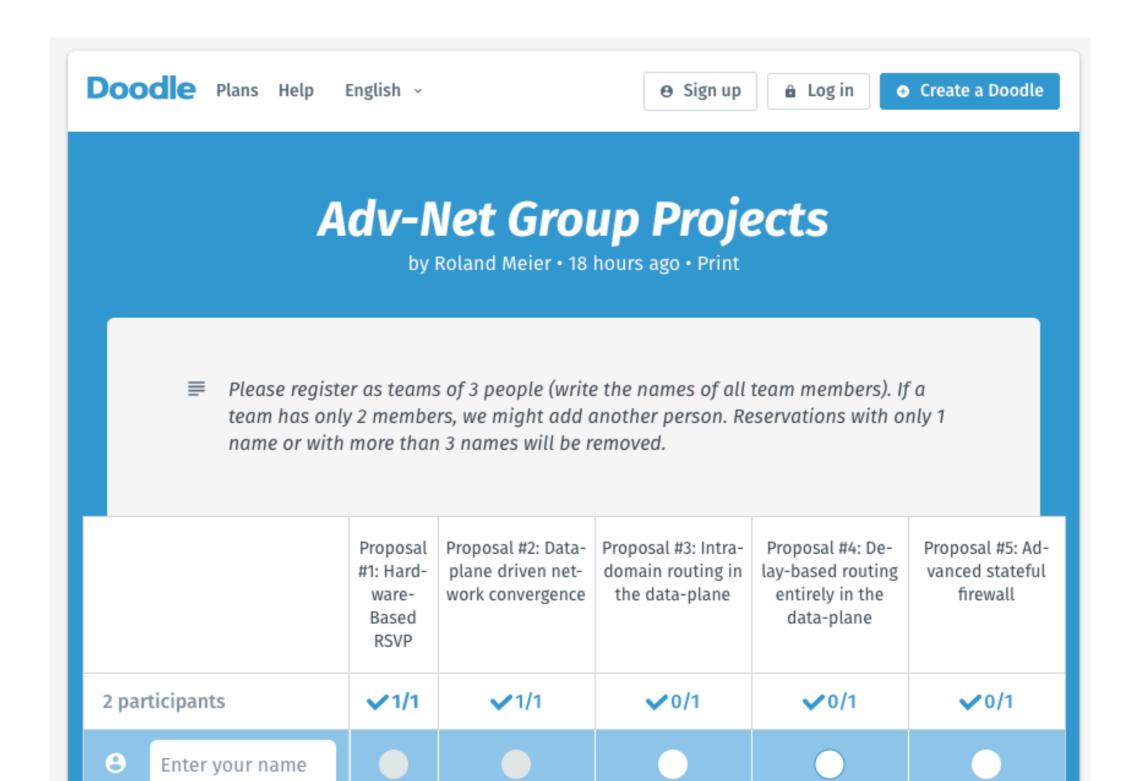
Responsible: Albert Gran Alcoz

Resource Reservation Protocol (RSVP) [1] is a signaling protocol that allows connections in a network to perform bandwidth requests throughout a given path. It is a protocol that has been included in different solutions both in the traffic engineering field and in quality of service. Integrated Services (IntServ) was the first in adopting it, in the late 1990s, as a means to provide guaranteed quality of service in multimedia networks. Some years later, and with higher success, RSVP was extended for traffic engineering purposes in the RSVP-TE protocol [2] to be used for the establishment of virtual circuits in MPLS. RSVP suggests users in a network to perform bandwidth reservations before starting data transmissions. For that, packet probes are forwarded from source to destination, letting routers in between identify the amount of resources requested by the new connection. Routers will receive those requests and reply to them by annotating in the same packet their resource availability. Flows will only be admitted if all routers along the path have agreed on having enough resources for hosting the new request. Although achieving notable and robust performance, being able to provide 100% resource guarantees, the high price that RSVP requires in terms of scalability and complexity, has made from it a not very successful solution in multiple scenarios until nowadays. Among the main drawbacks, the most remarkable ones are the time required to set up a new connection (too high especially for real-time flows), the amount of state to be stored in each switch along the path (to keep track of reservations), and the periodic overheads needed to refresh reservation requests.

In this project, we propose the design and implementation of an evolved version of RSVP, based on P4, to be run directly on hardware. We strongly believe that a signaling protocol executed at line rate in the data-plane can be quicker in deploying configurations and faster in reacting to updates.

Students are expected to come up with a data-plane implementation, aiming to overcome RSVP original

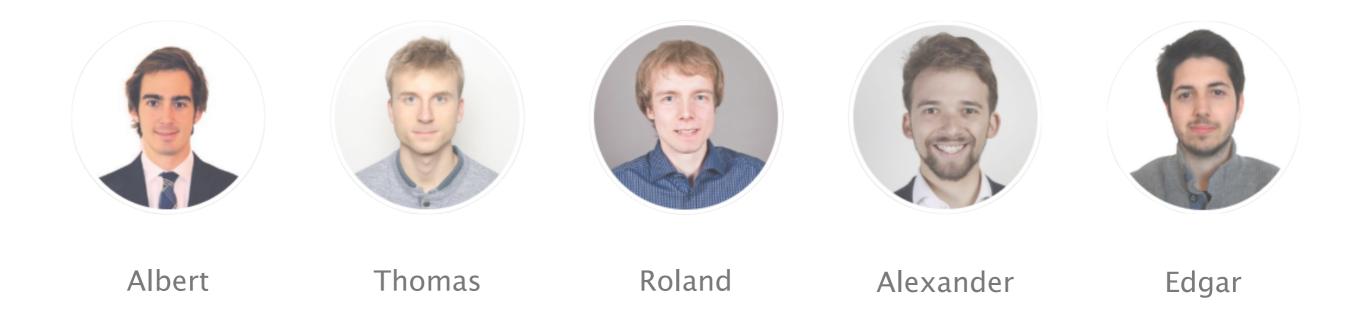
### Register your proposal (one per group) from Friday 3pm until Sunday 11.59pm



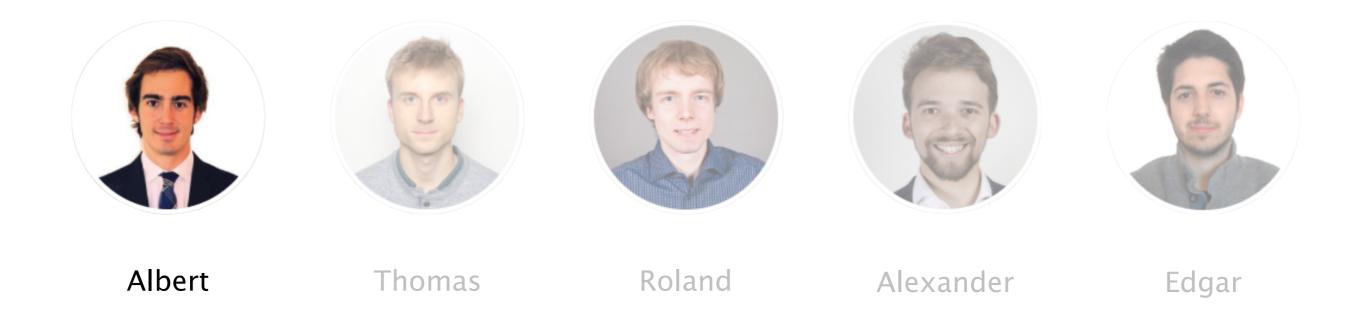
### If you want to propose your own project, send me an email describing it by Friday (Nov 2) 3pm

lvanbever@ethz.ch

### Quick overview of the proposals



### Quick overview of the proposals



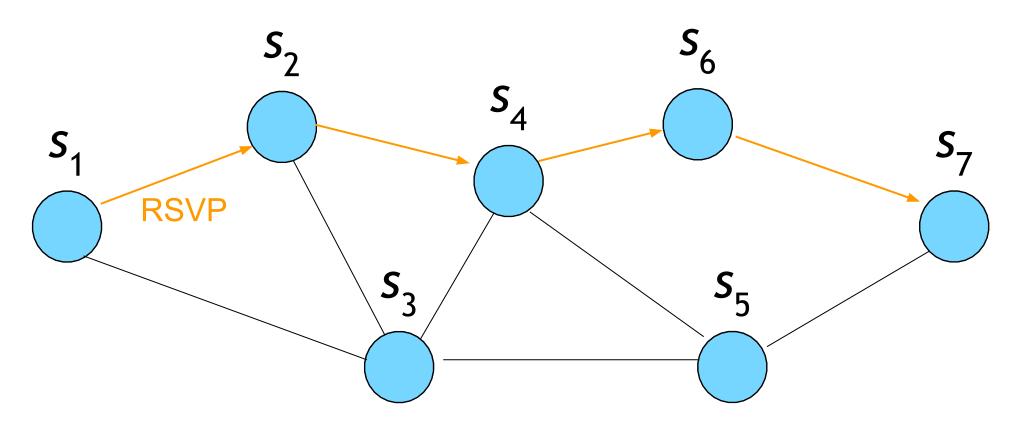
## Proposal #1 Hardware-Based RSVP

Bandwidth reservations throughout a given path:

- Quality of Service guarantees (IntServ)
- Establishment of virtual circuits (MPLS)

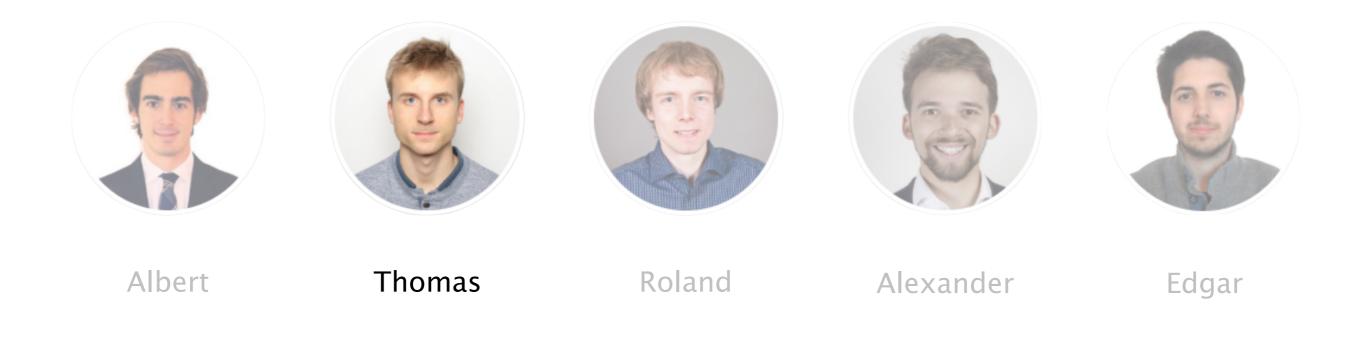
Exclusive data plane implementation:

- Personalized headers
- Header stacks
- Registers
- Bloom filters

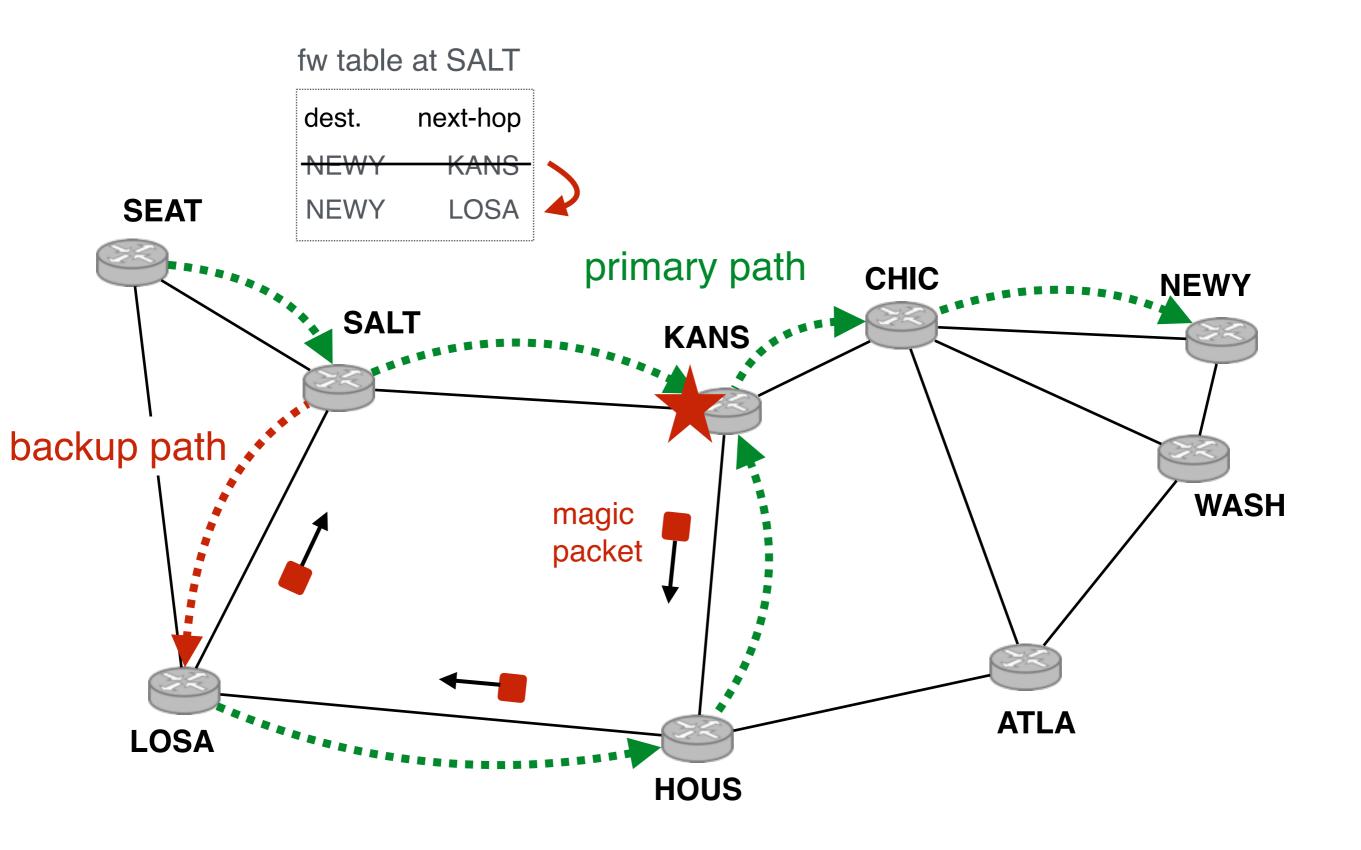


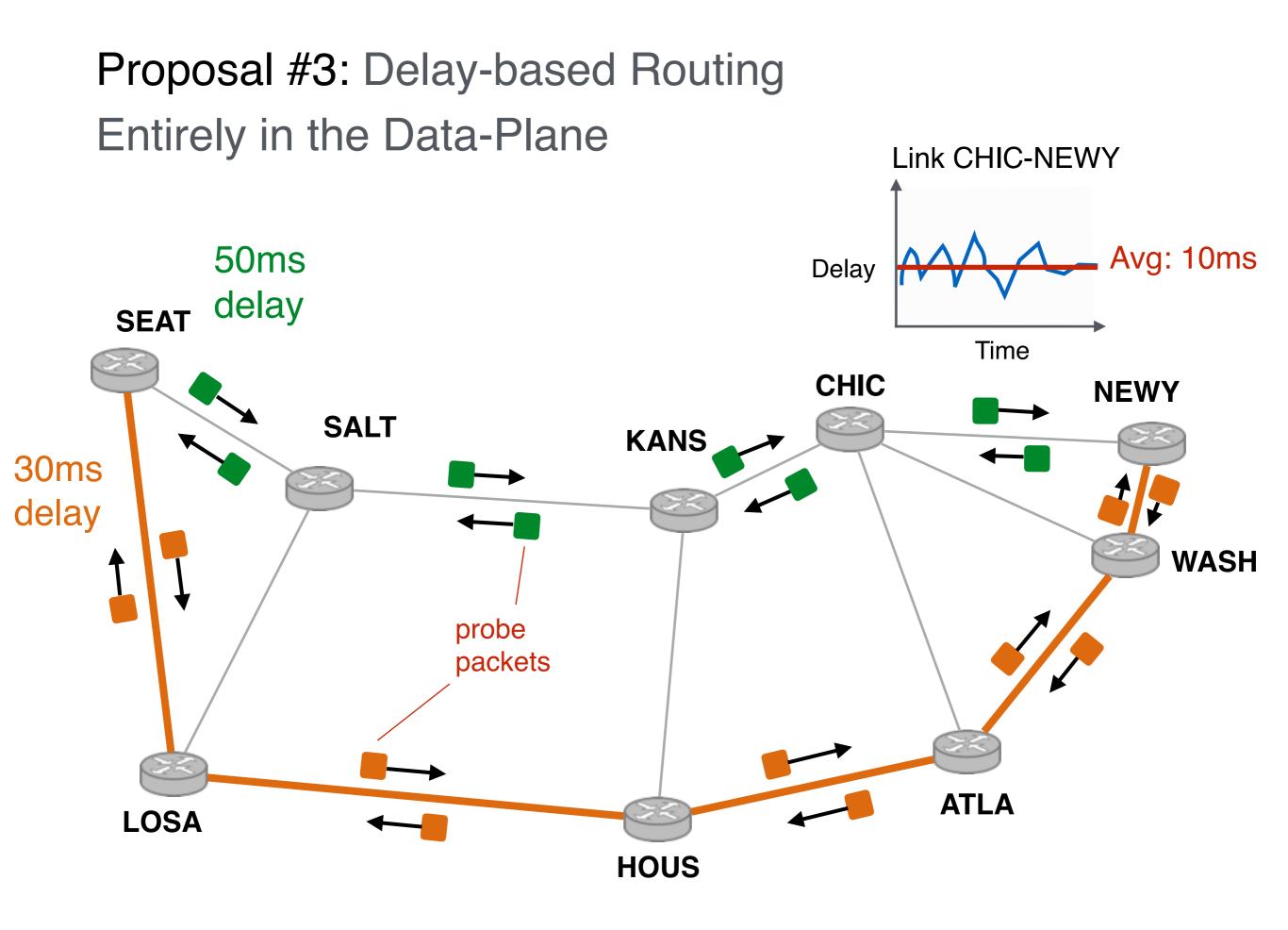
### Faster and more scalable than traditionally

### Quick overview of the proposals

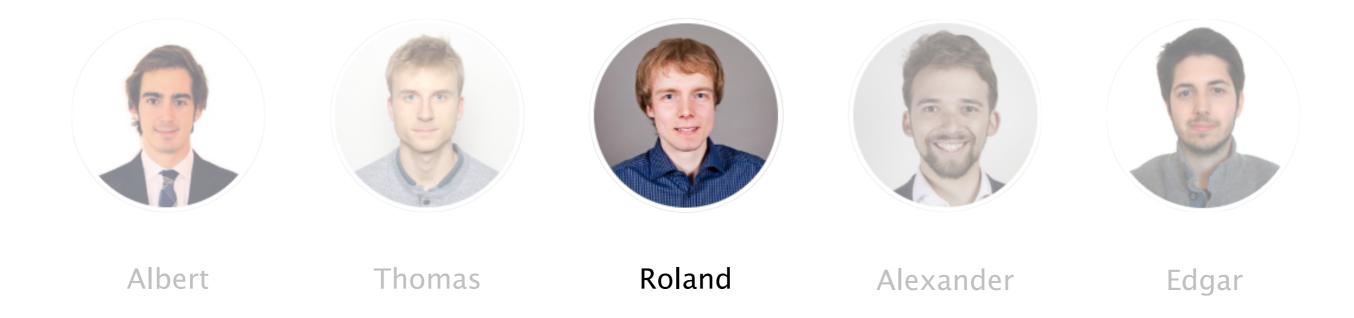


### Proposal #2: Data-plane Driven Network Convergence

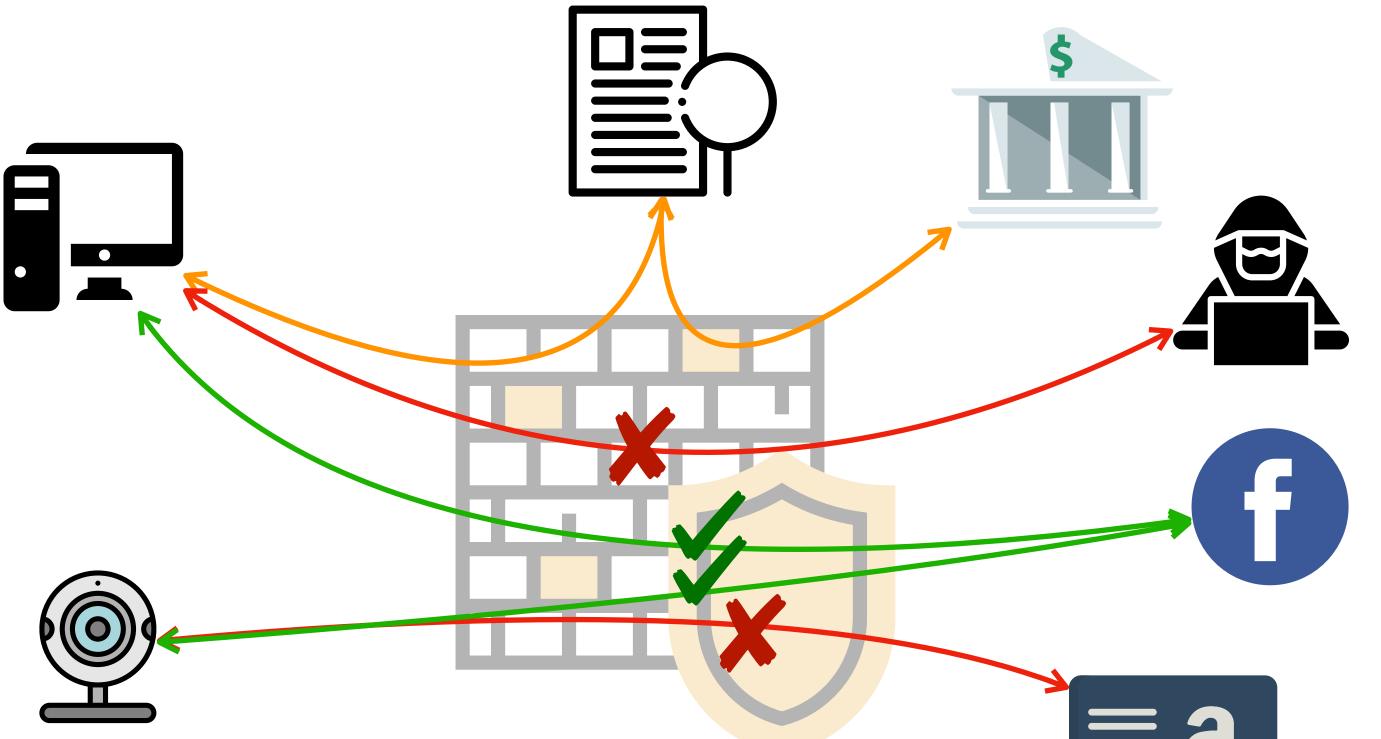




### Quick overview of the proposals



## Proposal #4 Advanced stateful firewall



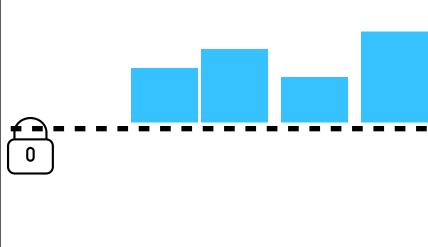
Fine-grained access policies
 Deep packet inspection (DPI)
 VPN
 Attack detection
 Spoofing detection
 (add your idea here)

. . .

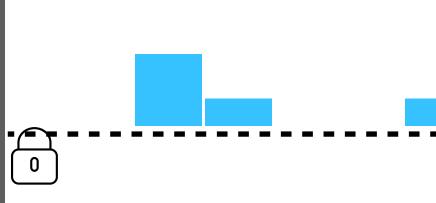


## Proposal #5 I know what you're seeing now

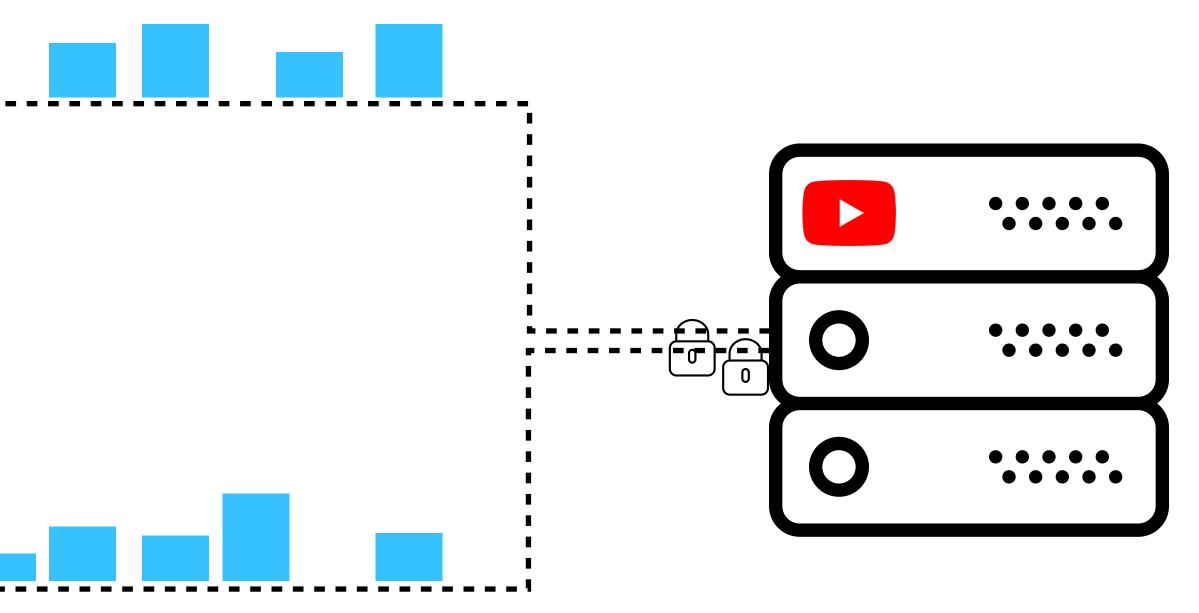






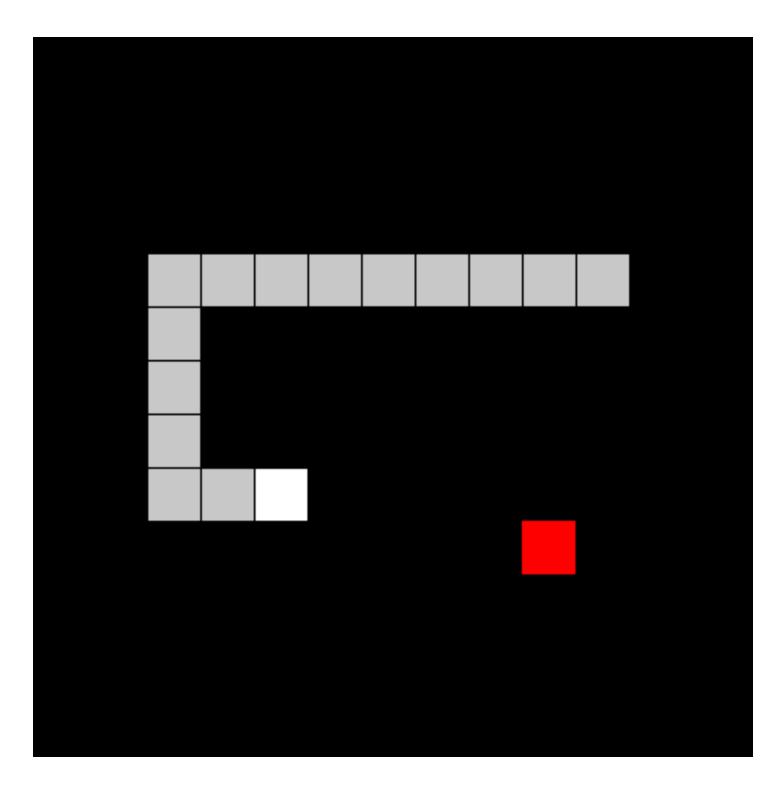


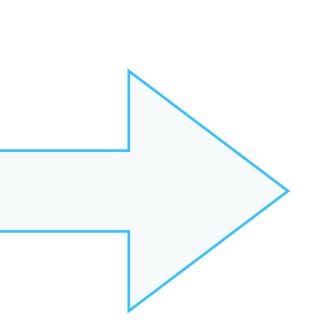


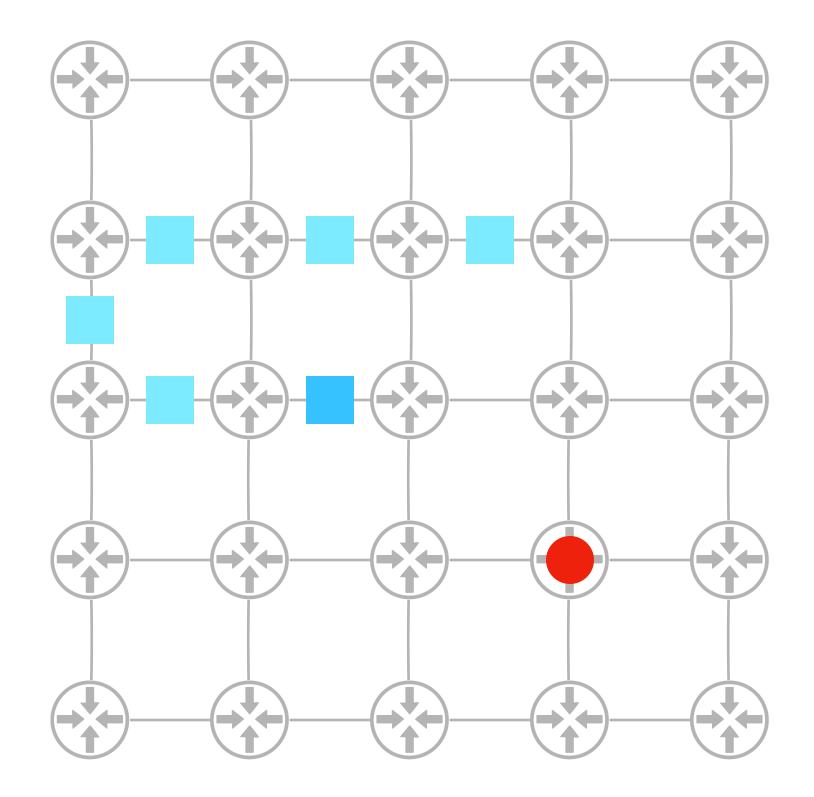




## Proposal #6 Playing snake in the data plane

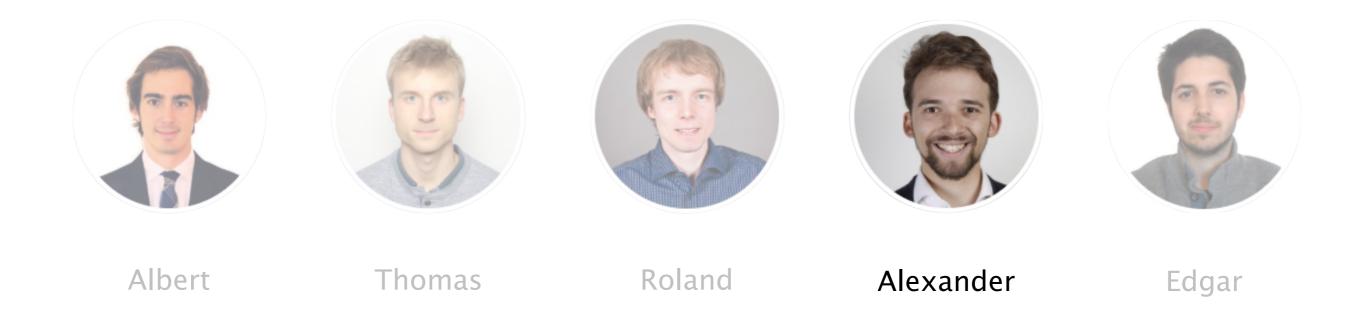






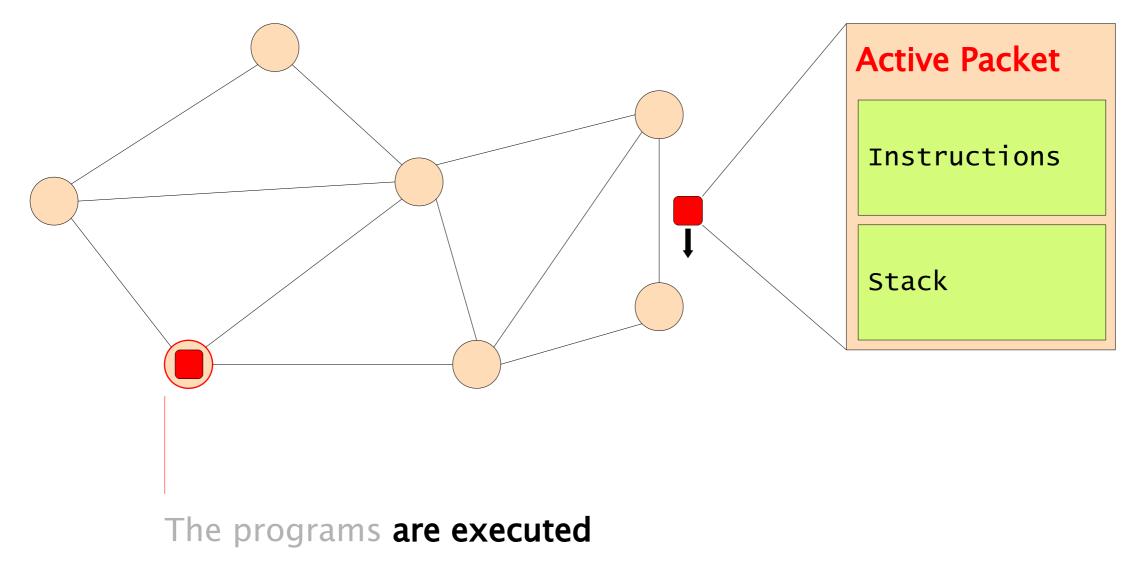


### Quick overview of the proposals



#### Proposal #7

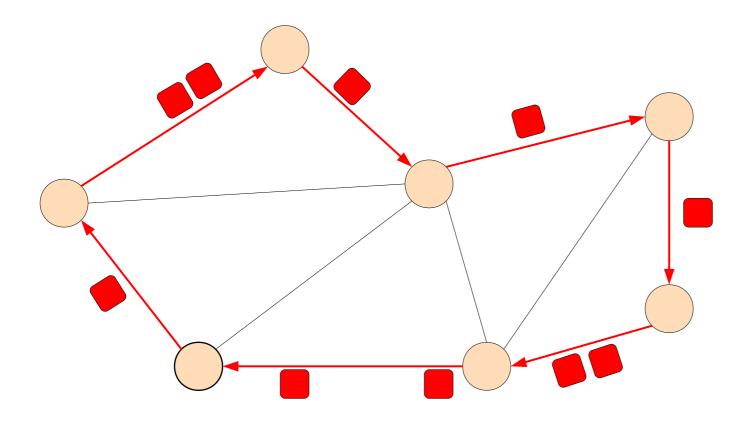
#### In Active Networks, packets carry programs.



on each switch along the path

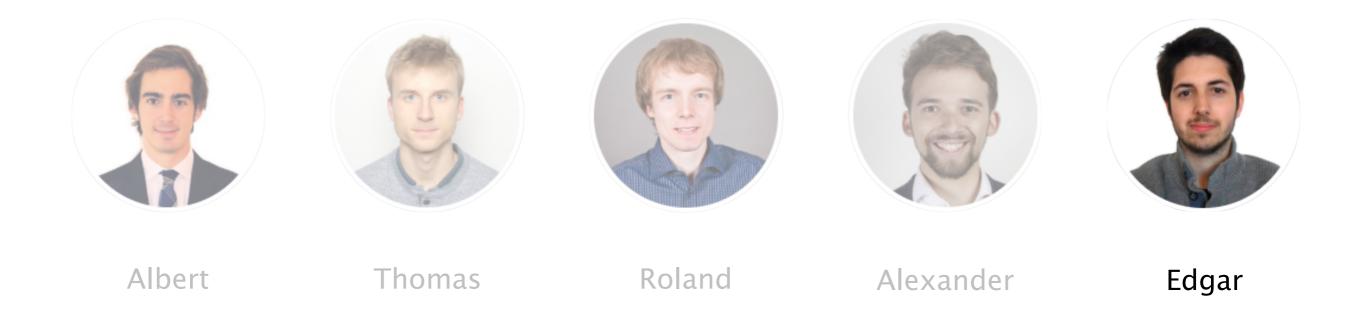
#### Proposal #8

### Storing data in the cloud the right way!

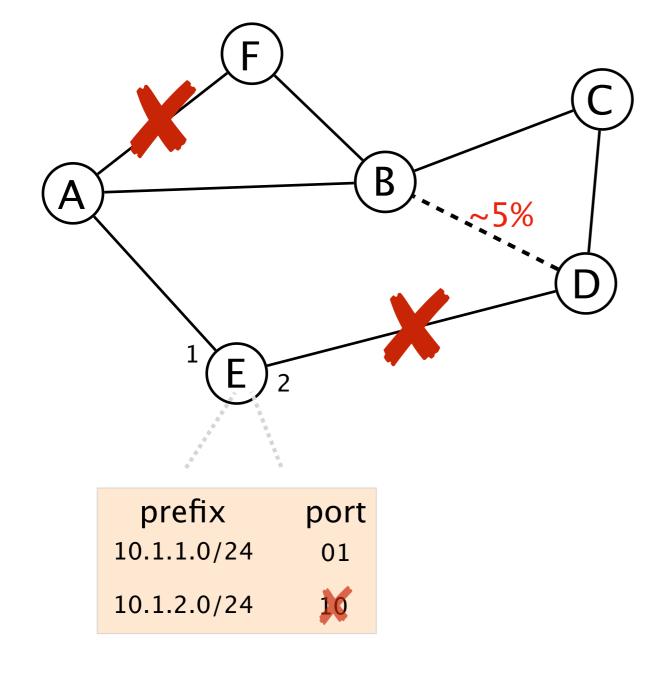


Store data in a forwarding loop

### Quick overview of the proposals



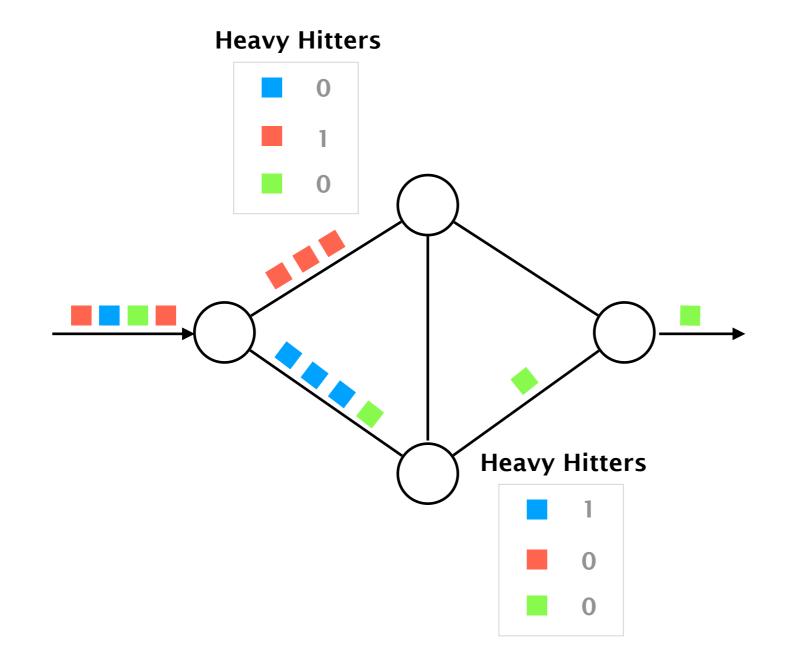
# Proposal #9 Data Plane Failure Detection

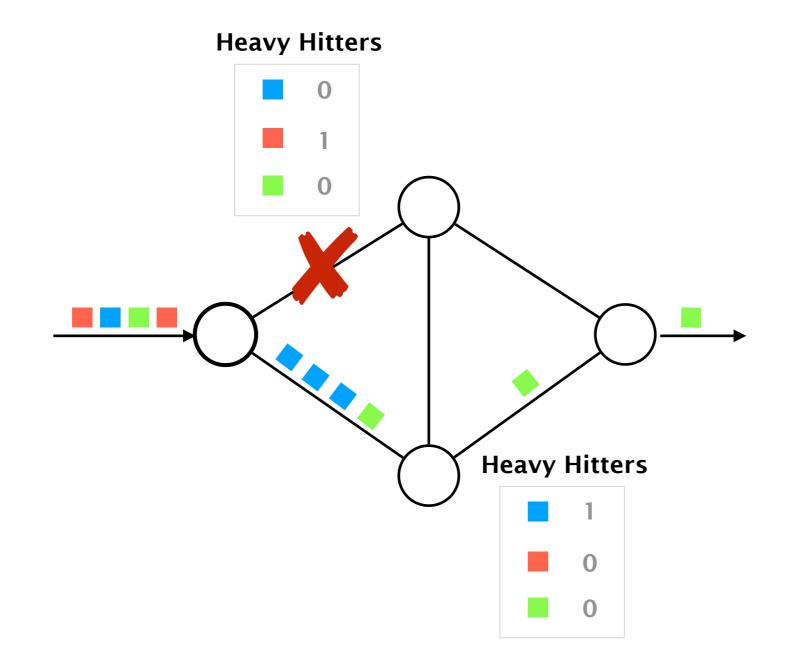


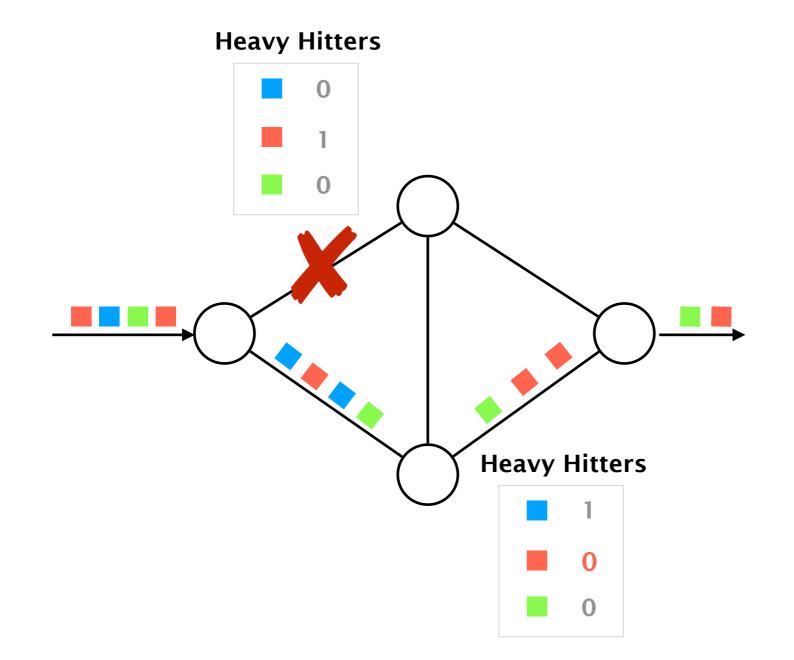
Detect local and remote link failures (A-C)

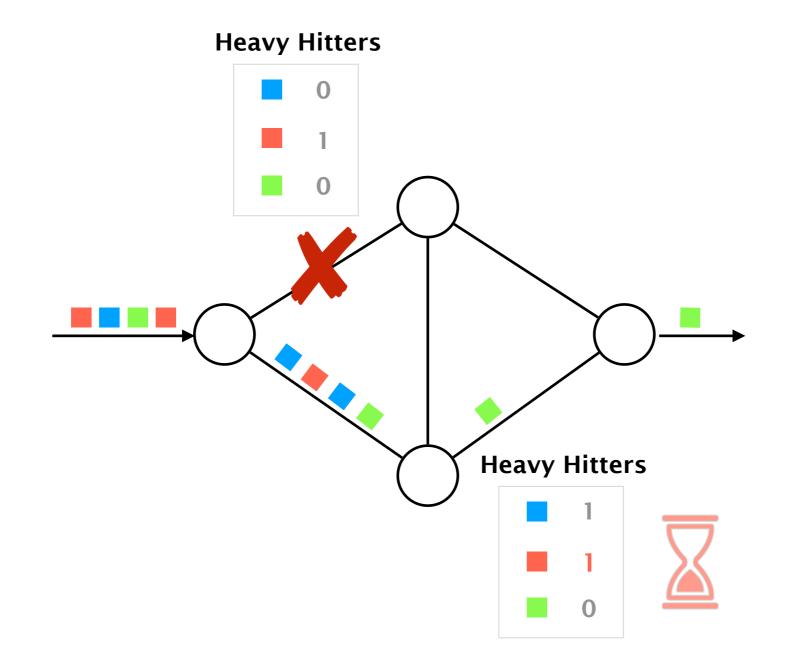
Detect random packet drops (B-C)

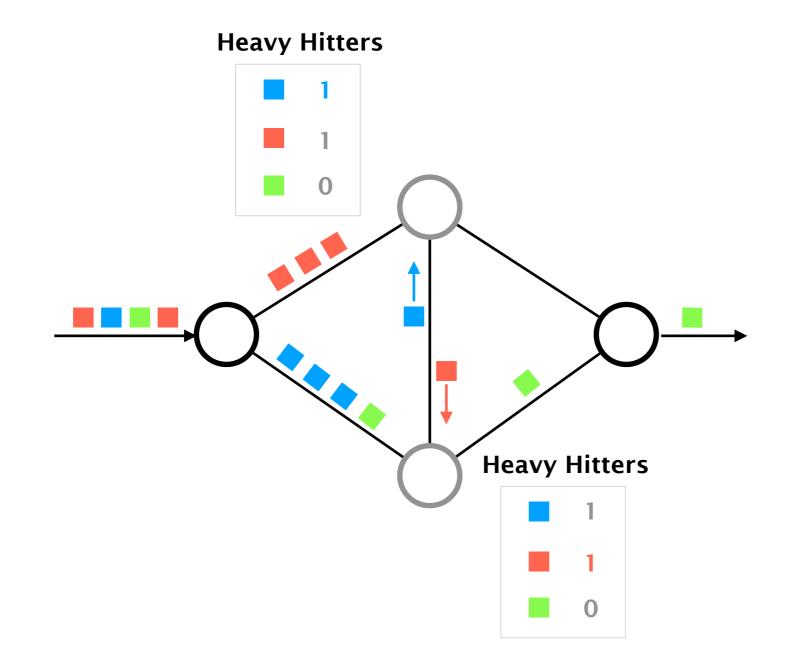
Detect corrupted table entries (E)











### Proposal #11

#### P4 Switch

#### Management and Configuration API

**Control Plane** 

#### **Basic Features**

I2 forwarding, learning, multicast ipv4, ipv6, I3 multicast ECMP, Weighted ECMP ICMP ARP ECN Simple QoS

#### Advanced Features

Spanning Tree Protocol netflow, sFlow or similar VXLAN, MPLS, Gre DHCP Server DNS Cache Simple Firewall NAT

Data Plane

# Advanced Topics in Communication Networks Programming Network Data Planes



Laurent Vanbever nsg.ee.ethz.ch

ETH Zürich Nov 1 2018

