A bit of history...
Programmable Networks
Network Programming

- High-level languages that provide general, reusable programming abstractions
- Compilers and run-time systems that generate efficient code for low-level devices
- Tools that automatically check key invariants
- Algorithms that offer robust performance and reliability guarantees
Applications

- Data Centers
- Enterprise
- Cellular
- Optical
- Wireless

- Routing
- Mobility
- Security
- Load Balancing
- Traffic Monitoring
ABSTRACT
This paper explores the possibility of implementing the widely deployed Paxos consensus protocol in network devices. We present two different approaches: (i) a protocol design which identifies a sufficient set of required OpenFlow extensions and (ii) an alternative, optimistic protocol which can be implemented without changes to the OpenFlow API, but relies on assumptions about how the network orders messages. Although neither of these protocols can be fully implemented without changes to the underlying switch firmware, we argue that such changes are feasible in existing hardware. Moreover, we present an evaluation that suggests that moving consensus logic into the network would reduce application message latency, and increase transaction throughput.

Categories and Subject Descriptors
C.2.1 Distributed Systems, Network operating systems, C.3 [Computer and Information Science Education]. Education, C.4 [Reliability, availability, and security]: Fault tolerance, C.2.4 [Computer and Information Science Education]. Applications

Keywords
Software-defined networking, Paxos, NetPaxos

1. INTRODUCTION
Software-defined networking (SDN) is transforming the way networks are configured and run. In contrast to traditional hard-coded network control interfaces, SDNs provide network devices using a set of protocols defined by open standards, including more prominently, the OpenFlow [24] protocol. This move towards standardization has led to new metrics such as "network programmability", allowing ordinary programs to manage the network through direct access to network devices.

Several recent projects have been initiated to explore how consensus logic can be moved into the network. In other words: how can distributed applications and protocols utilize network programmability to improve performance? Although none of this work has fundamentally considered whether application logic could be moved into the network. In other words: how can distributed applications and protocols utilize network programmability to improve performance? Although none of this work has fundamentally considered whether application logic could be moved into the network.

In summary, this paper makes the following contributions:

- Implementing Paxos in the network provides a different point in the design space, and identifies a different set of network requirements for protocol implementors.
- The paper presents two different approaches: (i) a protocol design which identifies a sufficient set of required OpenFlow extensions and (ii) an alternative, optimistic protocol which can be implemented without changes to the OpenFlow API, but relies on assumptions about how the network orders messages.

Although neither of these protocols can be fully implemented without changes to the underlying switch firmware, the paper presents an evaluation that suggests that moving consensus logic into the network would reduce application message latency, and increase transaction throughput.

In summary, this paper presents the following contributions:
Network Programming: A Grand Challenge
Why Us?
Cornell-Princeton

• Common research “taste”
• Rigorous and foundational
• Build prototypes and evaluate ideas in practical settings
Cornell-Princeton

• Interdisciplinary (Algorithms, Languages, Systems, Verification)

• Long history of successful collaboration (Frenetic, Pyretic, NetKAT, HotSDN/SOSR, etc.)
Our Goals

• Identify use cases and challenges
• Collaborate on research problems
• Tech transfer
• Student internships
• Possible affiliates program
Opportunity

• Build up a regional R&D community with a unique flavor
• Rigorous foundations
• Rich applications
• A “perfect storm” for great collaboration
Activities

• Meetings like this one (~1 / year)
• Tech talks delivered via “Google Hangouts on Air” (~4-6 / year)
• Blog (updated regularly)
Today

• Keynote talks by Ratul Mahajan (MSR) and Ben Pfaff (VMware)

• 15-minute “teaser” talks by faculty and PhD students on recent projects

• Panel discussions with you
Logistics

• Dinner at Dos Caminos (SoHo) @ 6:30

• WiFi: eduroam or Cornell-Visitor