Distributed Software-Defined Networking: The ACM PODC 2014 Workshop **DSDN**

Petr Kuznetsov[1]   Stefan Schmid[2]

[1] Télécom ParisTech
petr.kuznetsov@telecom-paristech.fr

[2] TU Berlin & T-Labs
stefan.schmid@tu-berlin.de

Abstract

The workshop on Distributed Software-Defined Networking, DSDN, took place in Paris, France, on the 15th of July, just before the 33rd ACM Symposium on Principles of Distributed Computing. The workshop intended to be a forum to discuss new algorithmic and distributed computing challenges offered by the emerging field of Software-Defined Networking (SDN). The workshop consisted of invited and peer-reviewed presentations, both from researchers in the field of distributed computing and in the field of networking.

1 SDN: Networking Is Cool Again!

Computer networking currently goes through a transition phase, and the paradigm of Software-Defined Networking (SDN) is discussed intensively, both in the industry and in the academia. In a nutshell, the paradigm out-sources and consolidates the control over a network to a logically centralized *software control plane*. This separation, and the introduction of "programmability", allows to adapt and innovate the network control plane more efficiently, and independently of the data plane. The resulting flexibilities open interesting new opportunities. Or, as the Cisco CTO recently put it: *networking is cool again!*

2 SDN meets PODC

At the heart of SDN lies the idea to design and operate the network control logic on a centralized network view. However, inevitably, this view is only *logically centralized* [5, 6, 8, 10]: in order to

avoid a single point of failure and ensure scalability and efficiency, the control plane state must be physically distributed.

The design of a distributed control plane is only one example where we feel that the PODC community could contribute to relevant networking problems today. Accordingly, the goal of the DSDN workshop was to bring together networking researchers with the PODC community, to discuss current trends in networking, and to identify interesting articulation points between the two communities.

The workshop took place in Paris on the sunny day of July 15th, just before ACM PODC, concurrently with four other PODC workshops.

3 Program

Below we give a short summary of the talks given in our workshop. Abstracts and slides can be found at `http://www.podc.org/podc2014/dsdn14/`.

3.1 Foundations of SDN

The opening keynote was given by Nate Foster (*Cornell University*), one of the leading figures in SDN and whose research is situated at the intersection of programming languages, networks, and security.

Nate gave an overview of the motivation for and foundations of SDN, and emphasized the importance that programming languages and formal methods play in software-defined networks: Only by a careful engineering of the right programming abstractions, an effective reasoning about network behavior becomes possible. The ability to formally reason about an SDN system is crucial, especially in the light of today's trend towards public cloud computing, where a network misconfiguration may leak confidential information to other tenants.

Nate presented his vision of the machine, language, and runtime models for SDN. In particular, he described the design of a machine-verified SDN controller, which is based on a detailed operational OpenFlow model and which is formalized in Coq. [4] This operational OpenFlow model can also be used to develop a verified compiler and run-time system for a more high-level network programming language such as NetKAT. [2]

3.2 Consistent Range Classification with OpenFlow

Yehuda Afek (*Tel Aviv University*) presented his recent work with Anat Bremler-Barr and Liron Schiff [1] on consistently managing flows and classifying ranges in SDN networks with multiple entrance, i.e., in scenarios where the flow changes the entrance point to the network. Their range classification scheme only requires three entries per range, and supports atomic updates across multiple switches. This makes the scheme attractive, e.g., in load-balancing and NFV applications.

3.3 SDN-Based Private Interconnection

Shlomi Dolev (*Ben Gurion University*) presented his joint work with Shimrit Tzur-David. Motivated by the advent of *hybrid clouds*, clouds augmenting private datacenters with the public clouds, Shlomi Dolev presented a new approach for private communication and data transfers leveraged by SDN. The work assumes that SDN can enable deterministic and manageable private virtual

networks between the local datacenters that reside in the private cloud, to the public resources in the public cloud.

In particular, Shlomi presented a private hybrid cloud in which all the information that passes across the cloud is information-theoretically secure, i.e., unless there is a sufficiently large coalition of malicious routers, the information cannot be revealed. The main idea of the approach is to use a secret sharing scheme together with SDN, to ensure privacy over multipath communication.

3.4   Software Transactional Networking: A Robust and Distributed SDN Control Plane

Marco Canini (*Université Catholique de Louvain*) presented the concept of software transactional networking (STN), a control-plane abstraction used for consistent composition of concurrent policies [3]. The abstraction assumes a set of control applications that concurrently apply modifications (or updates) of the *network policy*, i.e., the set of rules that stipulate how the traffic should be processed at the data plane. Given that the policy updates coming from different control applications may conflict with each other, the STN framework offers to the applications a *transactional* interface with *all-or-nothing* semantics. The appealing difference with classical transactional systems is that here we have to make sure that some kinds of transactions, namely *data-plane traffic* traces have to be processed in a manner that is transparent to network-configuration changes. The talk hinted on the formal definition of the abstraction of consistent policy composition (CPC), and sketched designs of its implementations that are resilient to asynchrony and failures in the control plane.

3.5   Declarative, Distributed Configuration

Can the challenges motivating the use of SDN be addressed using existing hardware and protocols? This question was asked by Sanjai Narain (*Applied Communication Sciences*) in his presentation. He summarized the experience collected at his company in using the Assured and Dynamic Configuration(ADC) system. In their approach, network functionality is expressed as a set of constraints on configuration variables. SAT or SMT solvers are used to convert these constraints into values of configuration variables. Of course, in specific scenarios, proprietary solutions may be more efficient and easier to deploy than generic ones (e.g., based on the SDN framework), and Sanjai's talk questioned the very motivation behind migrating to SDN.

3.6   Managing the Network with Merlin

How to program the network? Robert Soulé (*University of Lugano*) follows up on the question raised by Nate Foster at the beginning of the workshop, and points out that existing SDN languages focus mostly on packet forwarding and ignore other vital network features like bandwidth, packet processing, etc.

He then presented his network management framework Merlin, which later got accepted at ACM CoNEXT [11]. Merlin aims to simplify the task of network administration: it allows administrators to express network policy using programs in a declarative language based on logical predicates and regular expressions. Merlin comes with a compiler which automatically partitions programs into components that can be placed, i.e., *embedded*, on a variety of devices. It is based on a constraint solver to allocate resources such as paths and bandwidth.

3.7 Managing Dynamic Networks: Distributed or Centralized Control

In his somewhat provocative talk, Roger Wattenhofer (*ETH Zurich*) considers the following question: If we take the extreme case of a network managed (e.g., using the SDN approach) by a (fault-tolerant, performant, etc.) central controller, would this imply that distributed algorithms are no longer needed? The answer is no: we still have to deal with the problem of dynamicity and failures on the data plane, as well as the fact that the data plane is inherently geographically distributed and cannot be manipulated in the atomic manner. In short, we still have to deal with *consistency* of network control, and here we can benefit from distributed computing which is essentially all about consistency. In his talk, Roger overviewed several natural network consistency criteria (such as *loop-freedom* or *per-packet consistency*, see [9]) and sketched several impossibility results and complexity bounds of achieving these criteria in a few different network models.

4 Concluding Remarks: Distributed SDN? New and Interesting!

In a recent SDN survey [7], among the 400+ articles cited, we do not find a single paper that appeared in PODC or DISC. We think, based on our personal research experience, this is not because SDN does not give rise new and interesting research questions related to distributed computing. Maybe it is just lack of curiosity or insights on recent advances in networking?

We hope that the DSDN workshop could provide the networking and systems community with a glimpse into the world of PODC. We also hope that the workshop could help to demystify SDN, and inspire the PODC community to tackle some of the fundamental distributed and optimization problems offered by SDN. Indeed, the workshop shows that the situation is slowly changing: a few research groups are looking at the distributed aspects of SDN now. It seems, however, that progress here is still rather modest and typically boils down to solving "conventional" algorithmic problems, using SDN only as a motivating application. We believe nevertheless that there are deeper distributed and optimization challenges coming directly from SDN.

The good sign here is that the DSDN workshop gathered a good number of participants, fluctuated around 25-40 people, depending on the time-of-day, facing a strong competition with 4 concurrent workshops. For several DSDN participants, it was the first time to participate in a PODC event, and they used the opportunity to stay throughout the entire conference.

Overall, we are very happy with the outcomes of the DSDN workshop, and consider it a success.
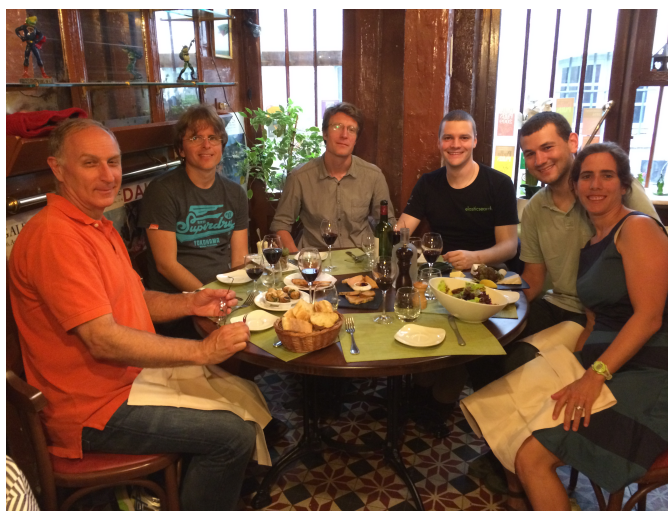
References

[1] Y. Afek, A. Bremler-Barr, and L. Schiff. Cross-entrance consistent range classifier with open-flow. In *ONS*, 2014.

[2] C. J. Anderson, N. Foster, A. Guha, J.-B. Jeannin, D. Kozen, C. Schlesinger, and D. Walker. NetKAT: Semantic Foundations for Networks. In *POPL*, 2014.

[3] M. Canini, P. Kuznetsov, D. Levin, and S. Schmid. Software transactional networking: concurrent and consistent policy composition. In *Proc. Workshop on Hot Topics in Software Defined Networks (HotSDN) (follow-up version: arXiv TR 1305.7429)*, 2013.

[4] A. Guha, M. Reitblatt, and N. Foster. Machine-verified network controllers. In *Proc. 34th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, pages 483–494, 2013.

[5] S. Hassas Yeganeh and Y. Ganjali. Kandoo: A Framework for Efficient and Scalable Offloading of Control Applications. In *Proc. Workshop on Hot Topics in Software Defined Networks (HotSDN)*, 2012.

[6] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama, and S. Shenker. Onix: A Distributed Control Platform for Large-scale Production Networks. In *Proc. USENIX OSDI*, 2010.

[7] D. Kreutz, F. M. V. Ramos, P. Verssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig. Software-defined networking: A comprehensive survey. Technical report, arXiv TR 1406.0440, 2014.

[8] D. Levin, A. Wundsam, B. Heller, N. Handigol, and A. Feldmann. Logically Centralized? State Distribution Trade-offs in Software Defined Networks. In *Proc. Workshop on Hot Topics in Software Defined Networks (HotSDN)*, 2012.

[9] R. Mahajan and R. Wattenhofer. On Consistent Updates in Software Defined Networks. In *Proc. HotNets*, 2013.

[10] S. Schmid and J. Suomela. Exploiting locality in distributed sdn control. In *Proc. Workshop on Hot Topics in Software Defined Networks (HotSDN)*, 2013.

[11] R. Soulé, S. Basu, P. J. Marandi, F. Pedone, R. Kleinberg, E. G. Sirer, and N. Foster. Merlin: A language for provisioning network resources. *to appear at ACM CoNEXT*, 2014.